CPC 464 USER

Issue number one: August September 1984

The Official AMSTRAD Micro Magazine £1.00



Comment Features Reviews Q/A

Amsoft

from the Source



The two official reference books that cover the BASIC and firmware of the AMSTRAD CPC464 are provided as concise technical specifications, complete with an update registration service.

These books are not tutorial works, and are intended for experienced programmers seeking a thoroughly indexed working reference. Written by the same people who wrote the software for the computer, there can be no more authoritative publication on the system. SOFT157 describes the BASIC in meticulous detail, including the screen and graphics handling extensions.

SOFT158 describes the firmware in over 450 pages: including machine code routines, jumpblocks, I/O and expansion philosophy etc. Both publications are supplied in ring binders with library slip cases.



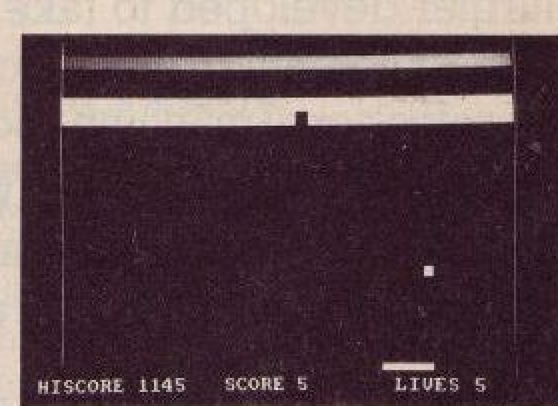
The Concise CPC464 BASIC specification
SOFT157 £19.95

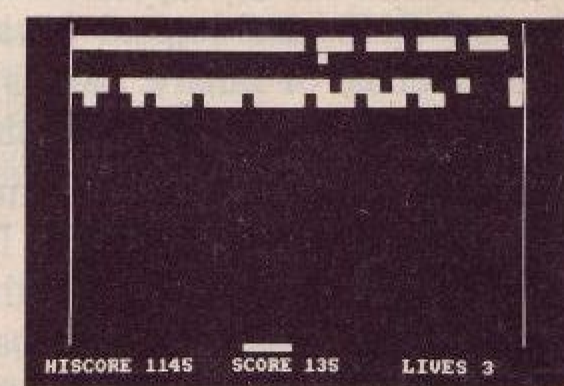
The Concise CPC464 Firmware Specification
SOFT 158 £19.95

CPC464 *USER*

Issue one, volume one

CONTENTS





Screen pictures above shows "Bustout" game, which amazingly uses only 42 lines of Basic

See page 28

Contributions are welcomed from all readers. If you want them returned, then please send a large SAE with all submissions.
Please address all correspondance to:

CPC464 User
Brentwood House, 169 Kings Road,
Brentwood, CM14 4EF, Essex

Consultant Editor Gareth Jefferson
Published by AMSOFT, a division of AMSTRAD Consumer Electronics plc.

HISOFT presents...

DEVPAC for the AMSTRAD CPC464

The complete relocatable Z80 assembler, disassembler and monitor has been further developed to take advantage of the attributes of the AMSTRAD CPC464. The 80 column display facility provides the user of DEVPAC on the CPC464 with a professional machine code development medium.

Devpac comes complete with a comprehensive line editor with renumber, block search and replace which speeds the development of assembly source files that may then be assembled at approximately 3000 lines per minute.

Expressions may contain addition, subtraction, multiplication, division, modulo, logical OR, logical AND and logical exclusive OR operators. Numbers may be specified as decimal, hexadecimal or binary. Labels of any length may be used (although the first 6 characters are significant). The symbol table size may be expanded as required.

GENA3 assembles all Z80 mnemonics, plus many assembler directives and commands, including ORG, EQU, DEFB, DEFM, DEFW, IF, ELSE and ENT. The assembler commands available are *H (heading for the listing), *L ('+' turns the assembler listing on, '-' turns it off), *S (pauses the listing), *E (new page - produce blank lines), *D (give addresses in decimal) and *F (assemble the source from tape, thus freeing the computer's memory for object code).

MONA3 combines a full Z80 disassembler and a powerful de-bugger which permits single stepping of machine code programs - even in ROM. There's a front panel display of the Z80 registers with their contents. The contents of a block of memory (which may be changed if required) - and a disassembly of the instruction on which you are currently working.

The commands available include the ability to convert decimal to hex, enter bytes directly into memory, enter ASCII into memory, fill a block of memory with a specified byte. You can display the alternate register set, move a block of memory, display a page of memory in hex with the ASCII equivalents, disassemble a block of memory to either a printer or a textfile of mnemonics that may be used by GENA3. You can search the memory for a given hexadecimal pattern, set multiple recoverable break points, continue execution from a particular memory location and single step instructions with a single keystroke - anywhere in memory.

DEVPAC is fully compatible with the proposed disc standard, and comes complete with a comprehensive instruction manual folder and library slip case.

Available from AMSOFT and good software retailers for £24.95 inc. (SOFT116)

EDITORIAL

No doubt after a lot of careful deliberation, you decided to buy the CPC464, affectionately known as the *Arnold*. We need hardly tell you that you made a wise choice. By now, at the very least, you will have run the demonstration tape and you will have seen for yourself just how powerful this amazing machine is. Not only is it powerful, it is by far the best value for money in computing today. These two facts will change the face of popular computing.

The unique combination of power and value is sure to make the Arnold one of the most popular computers in the country - in schools and offices, as well as in the home. But having bought the Arnold, you have acquired more than just a unique computer, because Amstrad has a commitment to every single CPC464 owner to offer a level of support unique in the industry.

Take this magazine, for example. It comes to you as part of the service offered to all User Club members, and it is our intention that it should entertain, inform and guide in a way that will help you get the very best from your purchase. User Club members are also entitled to special discounts on Amstrad computer products and software, and there's a regular opportunity to enter our software competition -see page 29 for details.

Something for everyone

When we started to think about this magazine, we realised that not everyone is a computer expert, so every issue will contain feature articles that will explain computer basics in an accessible, easy-to-read manner. We call these 'Level I' articles. Reading them should quickly get your knowledge up to the level where all today's computer jargon can be taken in your stride.

For people who know their ROMs from their RAMs but are not yet complete experts, we have 'Level II' articles designed to open up more of the secrets of the Arnold and computers in general. These articles will include features on how the computer works, improving your programming techniques and background stories.

At a higher level still, for people who are thoroughly at home with computers and who chose the CPC464 because of its technical superiority, we have 'Level III' articles to cover technical aspects in greater detail.

The Next Generation

Many schools already have computers; many more will have them now the CPC464 is available. We think the educational field is one of the most important aspects of microcomputing and we intend to reflect that in this magazine. Future issues will contain one or more articles devoted to computers in Education to show how our schools are using them and how you can help with your children's education too. Many educational programs are already available for the CPC464, and more are on the way. The educational pages will keep you up to date in this important area.

Feedback

One of the best ways you can help us to help you is to let us know your problems or suggestions so that we can air them in these pages. We'll try to answer your queries wherever possible and hope that this 'open forum' will be of benefit to all members of the User Club. This is the place where we can all help each other, so let's have plenty of letters please!

Articles

If you feel you have something useful to contribute that's too long for a letter, why not try your hand at writing an article for publication. We'll pay top rates for feature length articles accepted for publication, so your contribution could pay for your next software purchase.

Up-to-date

This is a fast changing world, but the magazine will keep members fully informed of all the latest products and developments. Get the best from your computer and read every issue.

CPC464 Launched in a Blaze of Publicity

Over 400 journalists packed the Great Hall of London's historic Westminster School, on April 10th and 11th to hear what Einstein, Ravel, Archimedes, Monet and Shakespeare had to say about Amstrad's new personal computer.



Messrs. Monet, Shakespeare, Ravel, Einstein & Archimedes

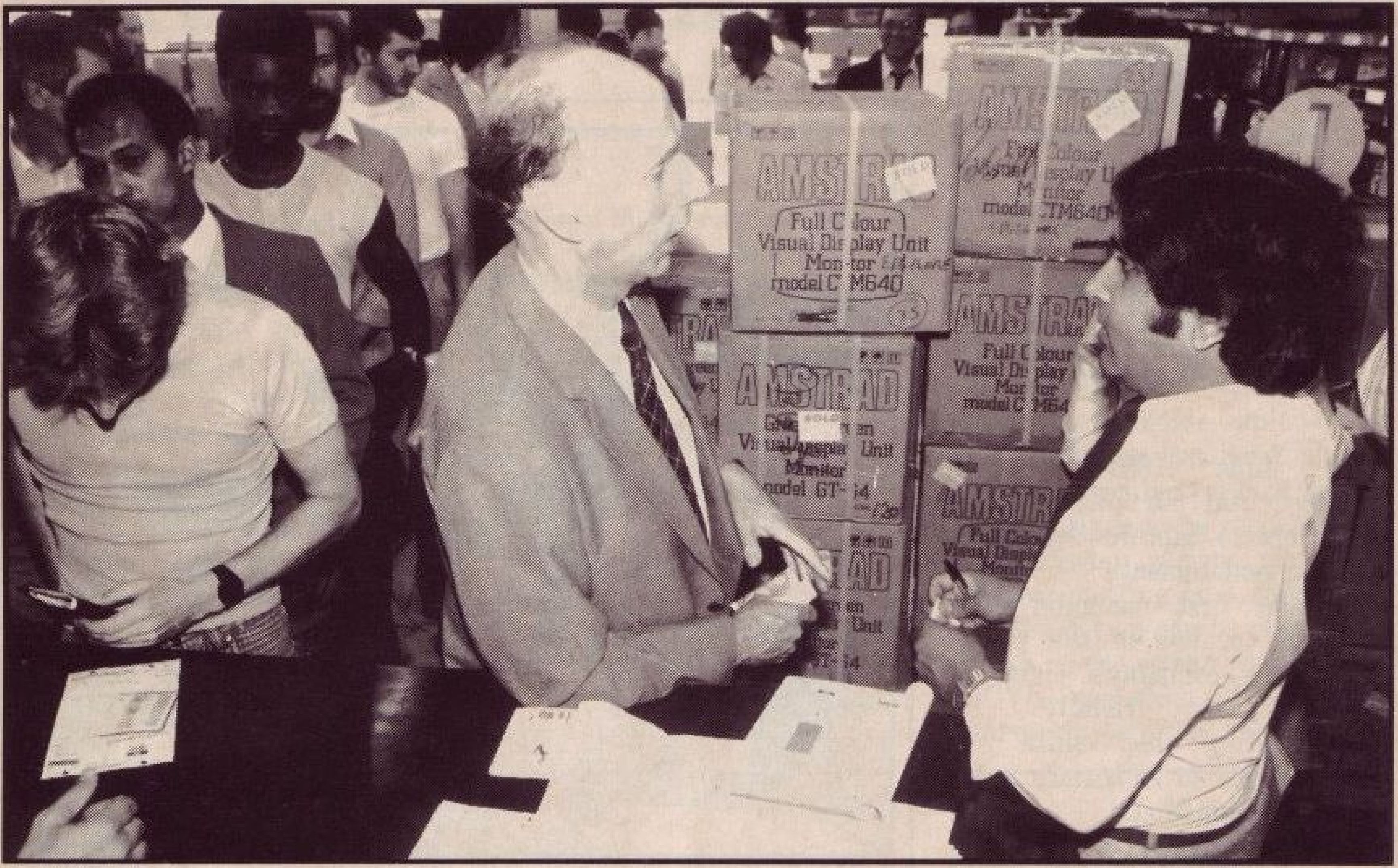
The namesakes of these very famous people had been found through extensive research and they were invited to demonstrate the CPC464's abilities.

They, each with the help of a CPC464 and some ingenious programming by Amsoft, performed admirably. Einstein proved that E did equal MC^2, Ravel played the Bolero from graphically displayed sheet music, Archimedes demonstrated a rather risqué version of the displacement of water, Monet showed a digitised picture of his namesake and Shakespeare showed how easy writing Hamlet could have been with a word-processor package. The audience was clearly impressed.

No sooner had Fleet Street gone to press, than the Switchboard was busy with customers eager to be pointed in the direction of their nearest Amstrad stockist.

At the press conference, Amstrad had indicated that supplies of the computer would be in the shops by the end of June. Bill Poel (General Manager of Amsoft) was so confident he was reported by Your Computer as saying "I will be prepared to sit down and eat one in Trafalgar Square if it's late". Despite the dubious but extensive publicity this would have achieved, Mr. Poel's gastric juices were not troubled.

First Computers On Sale at Rumbelows



Sixty people queued for nearly an hour outside Rumbelows in Edgware Road, London, eagerly waiting to be the first owners of the CPC464. As the doors opened at 9.30 on June 21st the crowd rushed forward.

In the first hour money and computers exchanged hands at an amazing rate. By 10.30 one hundred computers had been sold and only very few pieces of software remained. Customers had travelled from all over England and one gentleman even flew in from Bahrain with the specific purpose of collecting a CPC464.

What's in a Name?

"A rose by any name would smell as sweet..."

It's been called the Arnold by the team that developed it, the *Amstradivarius* by the Guardian, the *Mean Machine* by the Grimsby Evening Telegraph and the *Sinclair Beater* by Personal Computer World.

One trade magazine suggested that it was code-named *Amstar* and Computer News affectionately called it *Arthur*.

But of all the names the *People's Computer* seems to be most fitting. The Standard commented 'After the People's Car (the VW Beetle) the People's Computer'.

Using the Standard's analogy Amstrad is looking forward to worldwide sales of over 20 million computers, but like that other success story (the Model T Ford) in one colour only!

Your News Please

This news section will cover interesting and relevant items surrounding the CPC464 computer. It is designed to be your own user club newsletter and for this reason we welcome material from readers.

If you have a story, comment or point of view to make which you think might be of interest to your fellow readers then please write to me - Marc James - at Amstrad Consumer Electronics PLC, 169 Kings Road, Brentwood, Essex CM14 4EF.

Amstrad Goes to School

The value and performance of the CPC464 is the apple of teacher's eye in Thorpe Bay.....

Local School Headmaster Mr Sandford (of Thorpe Bay High School) had the foresight back in February this year to order 10 Amstrad computers. These were to equip the School's newly converted Computer Studies room.

On June 18th Amstrad delivered and our photographer was there to capture the school children's excitement.

As the first computer was switched on, one of the pupils proved the company's claim of being 'user friendly' by demonstrating his skills in programming to Amstrad's Sales Director Dickie Mould.

After a month of use and 'consistent daily hammering' as Mr Davis, the School's Deputy Headmaster put it, both the school and its pupils were obviously highly delighted. Mr Davis told us that the computers had been very well received - in his words 'there aren't enough superlatives to describe the CPC464, our pupils are deriving tremendous educational value from the Amstrad computer'.



Amstrad sales director, Dickie Mould, provides a personal delivery service for the first CPC464 systems to enter the educational system. There will be many more where these went...



Dickie Mould , Mr Sandford and Mr Davis await the reaction from the test market. Judging by the eager faces at the window, we shall one day be replacing keytops where the plastic has been worn out from over use!

Forthcoming Events

Amstrad will be exhibiting at the Personal Computer World Show at Olympia 2, London during September 19th-23rd 1984. You'll find Amstrad on Level 2 Stand number 1103.

On show CPC464s will be demonstrating the latest software together with printers, disk drives and many other peripherals.

See you there!

First impressions

Dave Atherton

I have had an Amstrad CPC464 for a few months now, as programmer on 'First Steps', the first BASIC tutorial book (with accompanying tape) from Amsoft and I'd like to tell you how I have found it. I have used many different micros, including the BBC Micro, and it was pleasant to find that the machine compared extremely favourably with virtually all other computers costing under £500.

MEMORY MAP

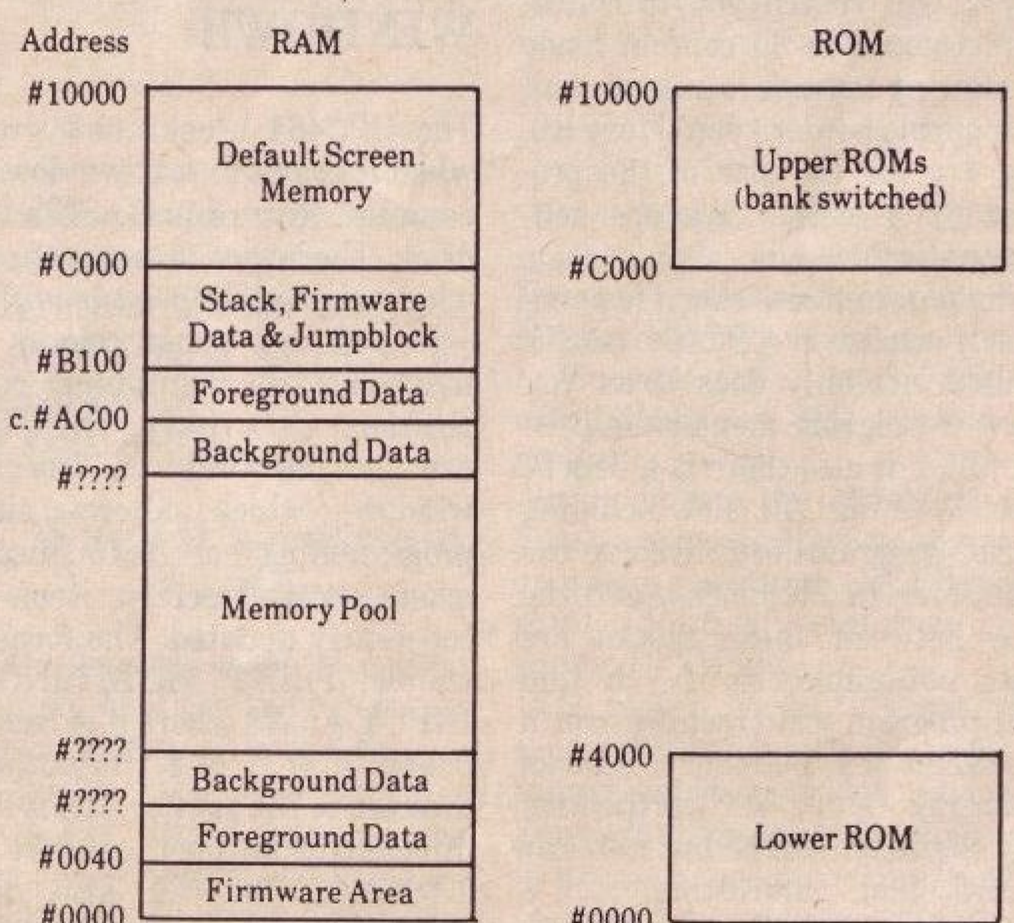
First of all, let's look at the memory map: Most of the locations documented here were discovered by experimentation and by looking at the documentation; they are by no means complete.

BASIC starts at &170 (& means hex) and finishes at HIMEM which defaults to &AB7F giving 42.5K for programs. Locations &BB00 to &BDF3, contain jump addresses to enter the ROM (see below) and &C000 to &FFFF are devoted to screen RAM in all modes. That accounts for most of the 64K map and you will find that miscellaneous system information is held in the 4K from &AB00 to &BB00.

So where's the ROM? Well, the clever thing is that the top and bottom 16K of RAM overlays 32K of ROM and is not easily accessible. PEEKs in BASIC and LDs in assembler read RAM contents normally and a latch has to be set to read ROM. However, this is not critical as all the routines that you are likely to need are available via the jump block mentioned above. Details of this jump block are to be published shortly. BASIC and the operating system also use the jump block so it can be treated as a set of vectors; user routines can be substituted if desired.

GRAPHICS

The graphics commands are very straightforward to use and follow the common principle of a graphics cursor. The three main commands are MOVE, PLOT and DRAW. Each takes a pair of X,Y co-ordinates and the usual effects happen: MOVE moves the cursor leaving no mark; PLOT gives a dot; and DRAW plots a line from the last cursor point. There are also MOVER, DRAWR and PLOTR commands which work relative to the last point plotted. This is particularly useful because small designs to be repeated as a line drawing can be done with relative plotting as a subroutine, and the design can be placed anywhere on the screen with a single MOVE command. These commands give the BASIC an



and & are used to indicate hexadecimal notation.

edge on ZX BASIC which only offers relative plotting, and the BBC Micro where only absolute plotting is supported by high-level commands.

FUNCTION KEYS

A particularly strong feature of the micro is the ability to program the numeric keys to contain command strings. The CPC464 BASIC does not support abbreviated or single-key BASIC, so frequent commands such as LIST and RUN can be set up on these keys that saves a lot of time in program development. The set-up that I have found most useful is:

```
key 0, "list"+chr$(13)
key 1, "run"+chr$(13)
key 2, "mode 2:ink 0,17:ink 1,0:
paper 0:pen 1:border 4"+chr$(13)
key 3, "renum"+chr$(13)
key 4,"?hex$(himem-fre(0)-&170,4)"+CHR$(13)
```

Key 2 will return you to convenient colours in 80 column mode for viewing listings (black on pink with purple border - you'll love it!). Key 4 gives the size of the program. The other keys are self-explanatory.

The program-size check is particularly useful in CPC464 BASIC because not only does it let you know if program memory is getting full, but also there is a benefit with SAVEing. All files including BASIC programs are saved on the datacorder in 2K blocks, and the gaps between these blocks are quite noticeable, so if you find your program just creeping over a 2K (&800 hex) boundary, a bit of squeezing may be worthwhile. Any character can be put into key strings; and, interestingly, if a control character is entered from the keyboard, the character appears as a graphic when the entry is made but acts as a control code when the key is pressed. The carriage return character is entered as chr$(13) above, although if the command ?chr$(1); chr$(13) is issued, then a return arrow will appear on screen. This can be COPYed into a key string, and will be treated as a CR character when the numeric pad key is pressed.

"A particularly strong feature of the micro is the ability to program the numeric keys to contain command strings"

WINDOWS

The CPC464 breaks new ground when it comes to text windows, an essential feature in BASICs these days. There are 8 separate text windows available each with its own pen and paper colours and independently maintained cursor position. CLS, LIST, INPUT etc can be directed to any text window, which allows simple programming of text displays where several screen items are constantly updated. The format is simple: PRINT #n,"STRING" or INPUT #n,A$ where n is between 0 and 7. If n is 8 the output is directed to the printer. If n is 9 it is directed to an open cassette file. Therefore you can also debug programs writing to printer or reading/writing cassette files by temporarily using a screen window. The screen windows are totally independent of each other and can overlap if required. There is often a case for using several windows, all set to a full screen just to maintain separate cursor positions.

CHARACTER SET

The character set is considerably more imaginative than usual there are 256 predefined characters and all 256 are redefinable! However, you lose 8 bytes from your 42.5K free, for each character that you define (except for the first 16). The designers have implemented all the block shapes you are ever likely to need, a Greek character set, games symbols (such as explosions, bombs etc.) and an assortment of shapes such as arrows, faces, ticks and crosses etc.

As all the characters are printable and definable a special arrangement has been made concerning the control codes -ASCII number 0-31 - covering the usual clear screen (12), bell (7) etc. This is done as follows: to print the graphics shapes associated with a control character, precede it with a CHR$(1). So PRINT CHR$(7) will give a beep but PRINT CHR$(1);CHR$(7) will print a space invader. Oddly, however, the design is such that pressing CTRL-G gives the space invader, not the beep. If characters above 32 are preceded with CHR$(1) there is no special effect. CHR$(1) also has a symbol associated with it, and to print it you have to issue PRINT CHR$(1); CHR$(1)

Q&A

QUESTIONS AND ANSWERS

What, you may well ask, is the first issue of a magazine doing with material for correspondance pages. Usually first issues are full of pleas for someone to write -not so CPC464 User -with the volume of correspondance being generated by the CPC464, we have a head start.

Here are some of the more frequently asked questions that we hope will satisfy the initial curiosity of the (already) many owners. We welcome all letters, but must point out that only letters accompanied by a stamped, self addressed envelope are likely to get a specific response. Many questions will be answerable by reference to the various publications and applications notes available from AMSOFT, and so we may simply direct attention to the relevent publication(s).

Much effort has gone into the initiation of documentation on the BASIC and operating system of the CPC464, and we sincerely hope that you will want to use this as your working reference.

Q. When is a pixel not a pixel ?

Dear Sir,

I have recently purchased a CPC464 and find its programming facilities excellent. It is so much better than the XXX I used to have - I have already joined the Club.

I have discovered a misprint in the manual not corrected in the 'Addendum'. On page 3, ch.5 where you describe the 3 modes, you say Mode 2's graphic resolution is 640 x 200, where it is really 640 x 400.

I have a few more questions about the machine: being interested in writing software such as games, I need to know how I can move shapes of any size around the screen. If this information is not readily available could you please suggest a book I could get.

Congratulations on bringing out such a good machine at such a low price.

OD, London NW3

A. When it's a screen address.....

The coordinates of points on the screen are kept the same between different modes, although the actual visible resolution changes. This means that a point on the screen has more than one address.

In mode two, there are 640 pixels, and 640 horizontal addresses. There are 200 vertical pixels and 400 horizontal addresses. So each pixel has two vertical addresses.

In mode one, there are 320 horizontal pixels, and 640 addresses. Here each pixel has two horizontal addresses. The vertical pixel count is the same for all modes: 200, so again there are 400 possible addresses.

In mode zero the 160 horizontal pixels are located by 640 addresses once again, so there are 4 possible points. Vertical resolution is as per modes 1 and 2.

The reason for all this is so that the expected aspect ratio is maintained between modes, and programs that use graphics which are written in one mode are readily transported to another. So it isn't so much a misprint as a misinterpretation. You'd be fairly hard put to spot the difference between between 400 and 200 vertical pixels ◊

◊ *on an average size screen anyway.*

Moving shapes around the screen falls into two categories: user defined characters being moved about in blocks by the `LOCATE` *command, or the more elegant but very much more complex question of individual pixel software 'sprites'.*

The first technique is relatively simple. Redefine the shape of some characters you don't need for the particular program using `SYMBOL`*, and place them on the screen using* `LOCATE`*. You cannot take advantage of the individually addressable pixel's colour using this technique, but it will quickly enable you to build complex shapes where you have the* `PEN` *and* `PAPER` *ink colours to work with (individually for each character cell if you want).*

To move areas around the screen and position them using pixel scrolling rather than character cell scrolling where each movement must be across a complete 8x8 cell matrix, you can use the `TAG` *command, although to obtain the speed and smoothness you see in games such as Electro Freddy, you must use software sprite routines. And here's a very large and potentially very intricate subject indeed that will feature in a future issue.*

Q. Tell me your secrets!

Dear Sir

From one of the machine reviews I have read, I understand that Locomotive Software are producing an extension manual for the computer. Is this so, if so, how much ??

As a professional hardware engineer, I am very interested in the expansion possibilities of the machine, and as a result I would dearly like a full circuit diagram of the machine.....can you please supply one? I would also like information regarding extension ROMs etc.

As I also write my own software (mainly for amateur radio applications), I am very interested in some of the routines used on the 'Welcome' tape for fast flashing, drawing etc. I have tried to break into the program, but you have done a good job with protection of the tape!

Is it possible to pass on any details as to how to list protected programs, I can assure you that such information will not be passed on to anyone else, as I wish to use the info for my own purposes only.

I hope that I am not presuming too much in writing this letter.

I would also like any info regarding the possibility of you marketing any suitable progs I may write for the amateur radio market.

RJT, Notts.

A. Yes and No!

Locomotive Software have produced two supporting publications which we have encouraged them to complete while the topic was fresh in their minds. The first is our SOFT 157, the Concise BASIC specification, which is a complete and detailed description of the BASIC interpreter and the various extensions for screen and sound.

The second is a rather more esoteric and advanced guide covering the operating system or firmware. SOFT 158 is a tome of considerable authority that describes the routines that are necessary to hold body and soul (hardware and BASIC) together. In fact, we believe that it is the most comprehensive system manual ever seen in this marketplace, and one which we hope will uncover even more of the storehouse of programmers delights than you might at first have expected.

A hardware service manual covering both of the monitors and computer unit is available from the Service Department of AMSTRAD. This is not a guide to extensions, but a complete guide to taking the machine to bits, and putting it back together. Price £5.30 inc.

Since much of subtlety in the hardware design lurks within the custom ULA, which contains a number of trade secrets, the user can get at the external bus connections - which include virtually all the signals necessary anyway. We are not able to assist with breaking the protection scheme, even if we knew how to do it ourselves. Some effort went into making the protection scheme more foolproof than most, and as an author of commercial software yourself, you should appreciate the dangers of copyright infringement.

As regards writing programs for AMSOFT, there are a number of points to bear in mind. The first is that AMSOFT presently encourages mainly general appeal entertainment software for mass merchandising purposes. Specialist software has a place with any machine, although the collation and marketing of this software is obviously more complicated than with the universal appeal of an arcade game.

To help users get around this problem, CPC464 User will be publishing a Program Marketplace for all CPC464 users where any programs can be 'advertised' under a number of classifications that will grow to suit the range of software on offer. Sales and wants will be covered.

There will be no charge for this directory service (yet!), and we will publish the following details:

Continued on page 21

PRINTING MONEY

Gareth Jefferson PRINTs #8 and LISTs a few tips...

A printer for your computer costs almosts as much as the computer itself, especially when that computer is as modestly priced as the Amstrad CPC464. Before you buy a printer, you have to ask yourself two vital questions: Is it a worthwhile investment? Is the printer I buy the best value for money?

Let's deal with the first question first. Irrespective of cost, a printer for your computer must be seen as a tool - a tool that helps you get the best value from your investment and a tool that turns your computer into a commercially viable product. Before you can justify the cost of a printer, you must analyse the ways in which you use your computer. Broadly speaking, there are three main areas of use for home microcomputers - games, software development and business use.

If you intend to use your computer only for games, you may not be able to make a strong case for buying a printer. But what a waste that would be! The games programs available for the CPC464 will provide hours and hours of entertainment and challenge, but that's only scratching the surface of the 464's capabilities. What do you do if you want to develop your own games?



(remember, there's a £2,000 prize for winning games entries - see page 29). What do you do if you want to use your CPC464 as a word processor using the AMS-WORD software package? Easy screen editing and formatting of letters and documents is not a lot of use without being able to print the results! What do you do if you want to make your business budgeting or domestic financial management more profitable if you can't keep a permanent record of the results? And then there's writing your own programs. All these activities need a printer. Let's see why.

Program development

Multi-million pound businesses have been based on the development of computer software, and Amstrad User Club members can also win big cash prizes for successful programs.

The CPC464 comes with an advanced version of BASIC, the world's most popular and easiest programming language. A powerful Z-80 assembler is available as a low cost option. So too is the famous Pascal programming language, favoured in leading universities and software companies throughout the world, and DR Logo, the 'artificial intelligence'

"Without a printer, you can forget serious software development, whether it's a commercial package for sale or a program you are writing for your own entertainment"

language that's paving the way towards truly intelligent computers. With these programming tools, the CPC464 owner has everything necessary to develop the computer software demanded by today's dynamic and expanding computer market. These tools allow owners of the Amstrad CPC464 to write software not only for the 464, but also for the many millions of other Z-80 based computers already in use world-wide.

Before you can reap these benefits, however, there is one prerequisite. A printer. Developing a fully 'de-bugged', working program involves a certain amount of trial and error. Even the world's most famous programmers - people such as Dr. Niklaus Wirth of Zurich who developed the Pascal language that forms the basis of Amsoft's HiSoft Pascal - cannot write programs that are 100% error free. To some extent, you can check your programs as you develop them on the computer screen. But, ultimately, you need a permanent record -hard copy in computer jargon - and that means a computer printout. Without a printer, you can forget serious software development, whether it's a commercial package for sale or a program you are writing for your own entertainment.

Printers and the Businessman

Businessmen tend to use computers either to help in financial projections, using spreadsheet programs such as Amscalc, or as word processors using word processor software such as Amsword. In both cases, paper printouts will be needed for permanent records.

The best type of printer for business applications depends partly on the nature of the task. Very high print quality may be needed in letters to certain clients, and in that case the printer will have to produce 'fully formed' letters in much the same way as a typewriter does. Computer printers of this type are called 'daisy wheel' printers. But daisy wheel printers are both slow and expensive, so most people opt for the faster, less expensive dot printers.

Dot printers have other advantages over daisy wheel printers apart from speed and price. Because the letters are formed from closely spaced dots, many models allow the printing of user defined graphics characters, graphs, pie charts and so on. These advantages are appreciated by businessmen who often want a permanent record of the computer's screen output (called a 'screen dump'). Ordinary daisy wheel printers cannot produce graphics and can only print the pre-formed characters on the wheel.

Word Processing

One of the most popular uses of home computers is for 'word processing'. Word processor systems comprise a computer plus word processing software that allows the computer to act as an intelligent typewriter. The computer keyboard is used like a typewriter keyboard, but the software allows the words in the document to be deleted, moved about or added to at will. Corrections and re-thinks no longer involve re-typing the whole document. Nearly all the articles you see printed in the computer press, including this one, were produced with the help of wordprocessor software, and businesses throughout the land, from solicitors offices to banks and estate agents, use word processors to increase their productivity.

Amsoft's Amsword allows you to do the same, but again one item is a must - a printer.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmno
pqrstuvwxyz1234567890¾=⅔½⅓¼%=÷=*"/$£_&'()|
²³#°<>@][!,.

ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmno
pqrstuvwxyz 1234567890!@£$%^&*()

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
1234567890!@$%£&'()_=£{}[]+;|:*,.<>/?\
```

Various styles of printout: 1)Daisy wheel printer 2)Philips GP300, which has many styles to choose from 3) Amstrad DMP-1

Continued on page 19

SOFTWARE REVIEWS

This month:

DEVPAC
Codename Mat
Amsgolf
Roland in the Caves
Roland on the Ropes
Oh Mummy!

Introduction

One of the surprising things about the CPC464 is the amount of software already written to support its launch. This stems from two reasons - the first is that a lot of effort has gone into introducing the machine to the software industry, both in terms of preliminary hardware, and also the copious documentary support for the BASIC and firmware.

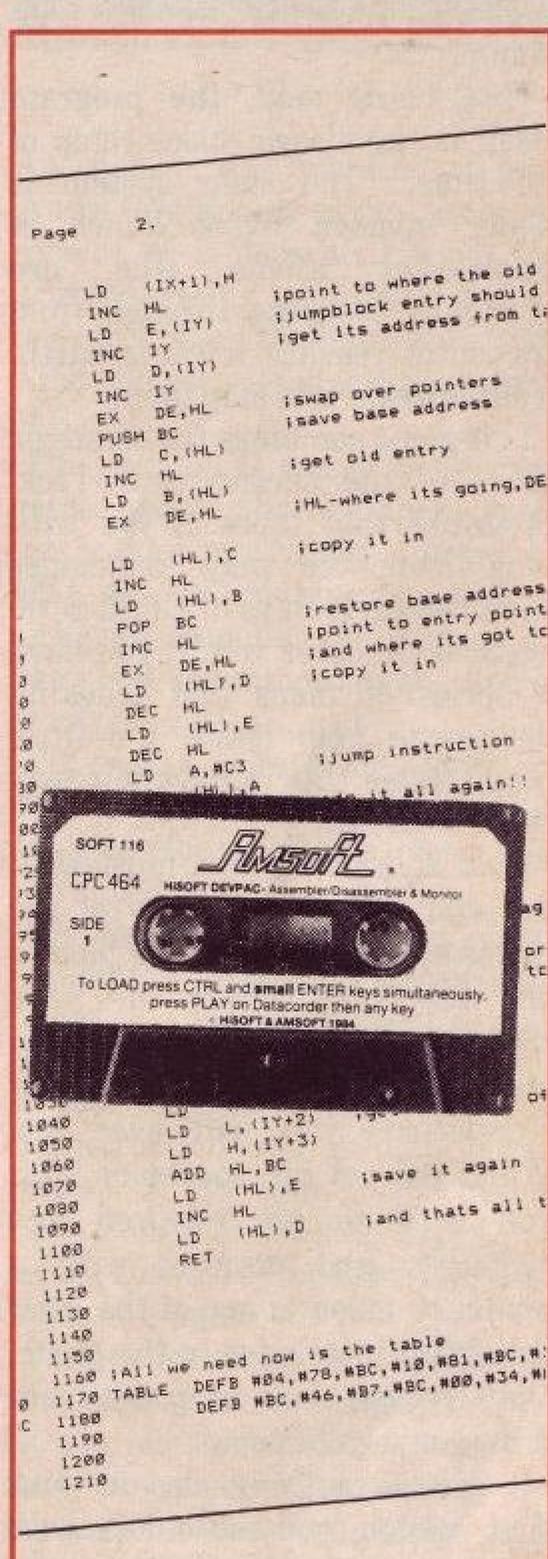
The second was the early development of a thorough assembler. HiSoft's DEVPAC full feature Z80 assembler and machine code development program was one of the first programs to be transported - a prerequisite for any machine code programming, so perhaps it's as well to review this product in some depth so that newcomers will see why it's such an important part of any software library.

To boldly LD where no man has LD'ed before....

Anyone daring enough to want to program in machine code is assumed to have some grounding in what it's all about - and since every book that's ever been written about machine code covers assembly language in some detail, it's not surprising to find the comprehensive and clear documentation of DEVPAC pitching users headlong into those delights of low level programming only available to those also equipped with a suitable reference on Z80 machine language.

An assembler is the medium through which assembly language programs are written and debugged - it isn't a mystical piece of software that somehow makes machine language programming as simple as BASIC.

With that in mind, DEVPAC provides just about every feature sought by the low level language programmer. And the beauty of HiSoft is that if you can think of something that you would like to see added or altered, then they will always consider constructive suggestions - which is certainly one of the contributory factors leading to DEVPAC's success on the Spectrum.



Code Name Mat
Micromega

With titles like Jet Set Willy, Manic Miner, Atic Atac, it's not surprising that the inspiration for games naming is on the wane. There was a time when Code Name Mat would have been called '3D-Hyper Mega Space Invaders' -or worse. As it is, Mat stands for 'Mission Alien Termination' - not exactly the sort of sentiment to broadcast in case we are Not Alone.

That being said, the program itself is the classic space romp of all time. The solar system is being invaded by a bunch of unpleasant aliens who are gradually creeping through from the outer reaches towards earth. Your mission is to stop them.

This game combines the strategic skills of the classic 'Star Trek' programs that those of you with long memories may remember from the early days of CP/M S100 systems. In other words, you have a series of maps and tracking devices to help locate the little blighters as they wend their way towards earth.

With these aids you must seek out and destroy as many of the aliens as possible (they are plotted on the Solar Chart for your convenience) by engaging them in one of the best 3D space fighter simulations you will ever see. When you get hit, sooner or later you need to get repaired and refuelled, and docking with a planet or moon is one of the most delightful interludes in the game. Don't get too close, or gravity will draw you to your doom.

If you do nothing else in your first session, you must dock with earth to see the detail that's gone into the game at every stage. Planets like Mars have their better known features on view -and although there isn't time or space to fit in all the known moons of planets like Jupiter, astronomers amongst players will not feel unduly cheated.

The only criticism is that the game takes a good evening to play properly. But if you've had enough of the ten minute wonders then this is a must. Originals of Code Name Mat may be changing hands in fifty years time in much the same way as the first editions of famous novels.

Micromega are further proof that for connoisseurs of good software that it's quality rather than quantity that counts in the end. If all they produce is one game every 3 months, and they're all as good as this, then you'll need three months before you are ready for the next one anyway!

Amsgolf
Computersmiths

Take heart all you BASIC programmers. This program was written entirely in AMSTRAD BASIC, yet you would never have noticed if I hadn't told you.

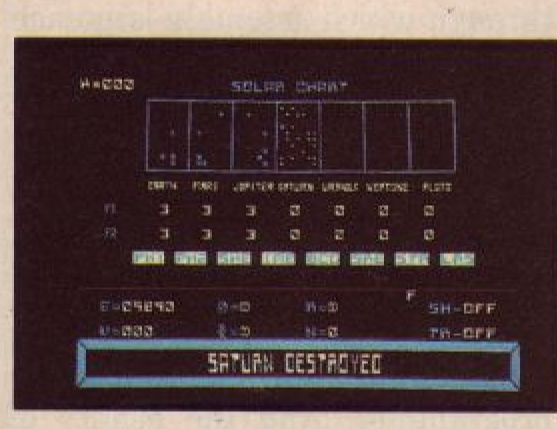
The game consists of playing on an 18 hole golf course, with the aim of reducing your handicap. You can't cheat when the program asks for your handicap, since you are only given the first password, and must earn the rest by playing under par, and then being given the next level keyword.

There's a decent range of clubs to choose from, realistic strength shots (ie you cannot drive 800 yards by over hitting). If you land in the bunker or on the green, suitable screens await you.

Bounce off the trees at your peril,

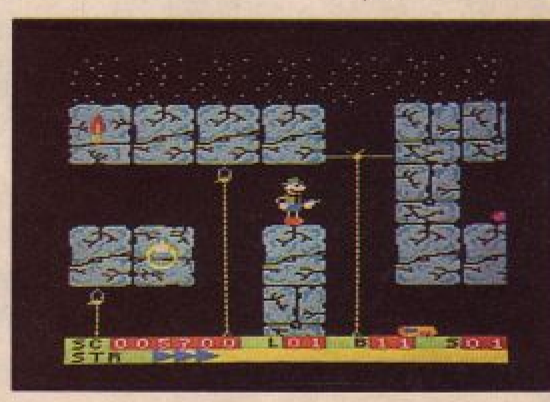












land in the water, in fact everything you might find on the local golf course without the expense or the exertion (or the danger of treading in the nasty reminders that the local dogs take their constitutionals on the course as well).

And if you wonder what the curious 'roar' is when you hit the green, that's only the crowd expressing their approval. Watch out for the 18th green, refreshment is at hand.....

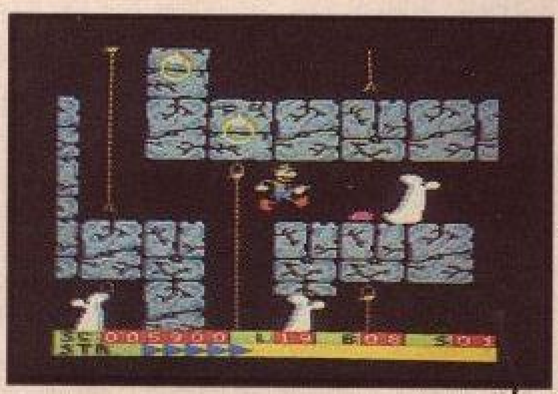
Roland in the Caves
Indescomp

One of the most original and brilliant games of all time has been further enhanced by the pixel precision of the CPC464. In the effort to scroll the most detailed screen of any CPC464 game, there is some evidence of the picture tearing at the edges - but this does not detract from the main action of the game.

The title screen is accompanied by a jaunty rendition of 'La Cukuracha' (I know it's not spelt like that, but when you hear the tune you'll recognise it immediately anyway!) The tune then gives way to exactly the sort of noise you'd expect to accompany a leaping flea.

Followers of games where virtually every key on the keyboard is needed will be surprised to learn that the only keys needed for this game are the Z and /. The idea is to guide the agile Roland (who bears a striking resemblance to a flea in this incarnation) up and out of the cave into which he has fallen.

Pressing the key for a brief time causes the hop to be restrained - although to get a full blooded leap you must time the depression carefully until the power has peaked, as indicated by a bargraph at the top left of the screen.



The only problems are the Roland-eating flowers that litter the cave ledges, and the hungry and ubiquitous pterodactyl whose unwelcome attentions you will do well to avoid.

One tip for the beginner - always start by jumping right, or you will end up leaping down the gizzard of the flying menace.

Roland on the Ropes
Indescomp

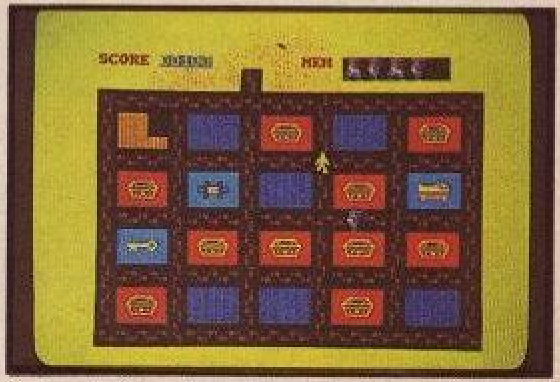
Another classic game with ghosts, spiders, rats, acid, treasure and some of the cutest and least menacing mummies you will stumble across. The idea is simple - get Roland from the bottom level of the dungeon complex (a random maze) to the top. Collect treasures, and avoid bumping into things that reduce his life expectancy.

The tune to send Roland on his way this time sounds rather like High Noon, although the nearest Roland gets to the OK Corral (I wasn't Film Buff of the Year either) is a trusty six shooter which can hold off the ghosts and zap the mummies.

Sound effects are convincing enough, especially the gunshots. The graphics are again an example to all in their use of colour and design, and you won't be surprised to learn that Indescomp are amongst the growing band of professional software houses who use trained graphic artists for design, and musicians for the sound track.

The game sets a high standard.

Oh Mummy!
Gem



Another in the fine line of programs that appear to the accompaniment of a demented tune. No prizes guessing what sort of tune to expect when the hapless British Archaelogist sets about the tombs of Egypt.

Oh Mummy is one of those rare games where the beginners level really does permit the unwary to beat the computer for once - although stepping up a grade or two gives the mummies the upper hand once again.

Why can't more games have an absolute novice level, or at least an option for unlimited lives while you hone the reactions?

Oh Mummy is a frantic dash around the catacombs, unearthing treasure, annoying mummies and eventually escaping with your life. As ever, Gem have provided pages of instruction and general background to the game. I wouldn't have been surprised to find Gem listing the inside leg measurement of the mummies.

Smooth graphics and accomplished sound, but programmable to be not too frantic to frighten off those newcomers who quake at the prospect of understanding the rules of some of the mega games, let alone actually playing them! You will be relieved to learn that the tune can be switched off before your senses are embalmed by its haunting melody.

The Surprise Listing !!

Every now and again we shall produce an anonymous listing with unforseen results. Here's the first. And don't complain to the editor if you reckon it's a waste of time, some people will type in anything...

```
10 INK 1,6: INK 2,24: INK 3,24: INK 4,1:  INK 5,1: INK 6,24: INK 8,1:I
NK 9,1: INK 10,24:INK 11,1: INK 12,1: INK 13,1: INK 14,24: INK 15,1
20 co=1
30 MODE 0
40 ORIGIN 70,0
50 GOSUB 1070
60 REM
70 REM load outline data
80 READ a,b
90 IF a<>999 THEN 130
100 a=b
110 READ b
120 MOVE b,a*2
130 IF a = 1000 THEN 160
140 DRAW b,a*2,1
150 GOTO 80
160 READ aa,ab,ac,y
170 FOR s= 1 TO 13
180 READ x
190 MOVE x,y*2
200 DRAWR 0,-(aa*2)
210 DRAWR ab,0
220 DRAWR 0,aa*2
230 NEXT s
240  REM
250 RESTORE 650
260 READ b,a
270 IF b<>999 THEN 310
280 b=a
290 READ a
300 MOVE a,b*2
310 IF b =1000 THEN 800
320 DRAW a,b*2,2
330 GOTO 260
340 DATA 999,2,28,2,464,12,464,12,28,2,28,999,14,56,18,56,18,436,14,43
6,999,86,56,86,436,102,436,102,56,86,56
350 DATA 999,12,36,134,36,134,44,12,44,999,18,72,86,72,999,18,420,86,4
20
360 DATA 999,134,80,134,28,140,28,140,88,136,96,124,96,124,84,132,84,1
34,80
370 DATA 999,134,28,130,20,130,8,134,0,140,0,144,8,144,20,140,28,999,1
38,20,136,20,134,16,134,12,136,8,138,8,140,12,140,16,138,20
380 DATA 999,96,80,92,80,90,76,90,68,92,64,96,64,98,68,98,76,96,80,999
,96,428,92,428,90,424,90,416,92,412,96,412,98,416,98,424,96,428
390 DATA 1000,1000
400 DATA 64,12,64,86,96,120,144,168,192,216,240,264,288,312,336,360,38
4
410 INK 1,6: INK 2,24: INK 6,24: INK 12,1: INK 8,1: INK 3,24
420 INK 4,1: INK 10,24
430 INK 13,1: INK 15,1
440 INK 9,2-ABS(co-1) :INK 5,2-ABS(((co+1) MOD(3))-1):INK 11,2-ABS(((c
o+2) MOD(3))-1)
450 co = (co+1) MOD(3)
460 FOR x= 1 TO 150:NEXT x
470 INK 13,14
480 INK 9,2-ABS(co-1) :INK 5,2-ABS(((co+1) MOD(3))-1):INK 11,2-ABS(((c
o+2) MOD(3))-1)
490 INK 2,1: INK 4,24
500 INK 10,1:INK 12,24:INK 3,6: co=(co+1) MOD(3)
510 FOR x= 1 TO 150:NEXT x
520 INK 9,2-ABS(co-1) :INK 5,2-ABS(((co+1) MOD(3))-1):INK 11,2-ABS(((c
o+2) MOD(3))-1)
530 INK 15,14
540 INK 4,1: INK 8,24
550 INK 6,1:INK 10,24
560 co = (co+1) MOD(3)
570 FOR x= 1 TO 150:NEXT x
580 INK 9,2-ABS(co-1) :INK 5,2-ABS(((co+1) MOD(3))-1):INK 11,2-ABS(((c
o+2) MOD(3))-1)
590 INK 13,1
600 INK 8,1:INK 4,24
610 INK 6,24:INK 10,1
620 FOR x= 1 TO 150:NEXT x
630 co = (co+1) MOD(3)
640 GOTO 410
650 DATA 999,134,240,134,224,132,220,132,216,126,204,124,204,122,200,1
14,200,112,204,110,204,104,216,104,220,999,104,252,104,256,106,260,108
,260,110,264,118,264
660 DATA 110,264,108,268,106,268,104,272,104,276,999,104,308,104,312,1
10,324,112,324,114,328,122,328,124,324,126,324,132,312,132,308,134,304
670 DATA 134,292,999,120,232,116,232,999,118,228,118,236,999,120,300,1
16,300,999,118,296,118,304
680 DATA 999,142,240,164,240,164,232,174,232,174,240,182,240,182,296,1
42,296,154,320,182,320,194,296,194,244,180,216,154,216,142,240
690 DATA 999,140,240,140,216,134,204,132,204,130,200,128,196,126,196,1
22,188,120,188,118,184,116,180,114,180,108,168
700 DATA 98,168,98,200,106,200,106,168,999,98,176,102,176,999,102,184,
98,184,999,102,192,98,192
710 DATA 999,140,248,140,284,999,140,292,140,316,136,324,134,324,130,3
32,128,332,124,340,122,340,120,344
720 DATA 116,344,114,348,112,348,108,356,98,356,98,324,106,324,106,356
,999,102,348,98,348,999,98,340,102,340,999,102,332,98,332
730 DATA 999,174,268,174,256,166,256,166,264,172,264,172,260,168,260,9
99,174,288,174,276,166,276,166,284,172,284,172,280,168,280
740 DATA 999,162,264,162,276,158,276,158,268,999,154,276,150,276,150,2
64,154,264,152,268,152,272,154,276
750 DATA 999,150,224,164,224,999,170,224,184,224,999,184,232,152,232,9
99,148,232,146,232
760 DATA 999,182,240,186,240,999,190,240,192,240,999,194,248,182,248,9
99,194,256,186,256,999,194,264,192,264,999,188,264,182,264
770 DATA 999,194,272,182,272,999,194,280,188,280,999,184,280,182,280,9
99,182,288,190,288,999,194,296,182,296
780 DATA 999,190,304,180,304,999,174,304,156,304,999,152,304,146,304,9
99,150,312,162,312,999,168,312,186,312
790 DATA 1000,1000
800 RESTORE 650
810 PRINT CHR$(23);CHR$(3)
820 READ b,a
830 IF b<>999 THEN 870
840 b=a
850 READ a
860 MOVE a,(b*2)-40
870 IF b=1000 THEN 940
880 IF b*2-40 < 208 THEN 910
890 DRAW a,(b*2)-40,4
900 GOTO 820
910 READ b,a
920 IF b =999 THEN 840
930  GOTO 860
940 RESTORE 650
950 READ b,a
960 IF b<>999 THEN 1000
970 b=a
980 READ a
990 MOVE a,(b*2)-80
1000 IF b=1000 THEN 400
1010 IF b*2-80 < 208 THEN 1040
1020 DRAW a,(b*2)-80,8
1030 GOTO 950
1040 READ b,a
1050 IF b=999 THEN 970
1060 GOTO 990
1070 SYMBOL 244,96,248,252,254,126,63,15,0
1080 SYMBOL 249,48,120,120,124,60,30,14,6
1090 SYMBOL 247,6,15,31,30,62,60,120,32
1100 SYMBOL 246,4,14,15,31,62,60,112,192
1110 SYMBOL 245,24,24,60,126,60,24
1120 PEN 13
1130 LOCATE 7,12:PRINT CHR$(249)
1140 LOCATE 15,12:PRINT CHR$(247)
1150 PEN 15
1160 LOCATE 3,11:PRINT CHR$(244)
1170 LOCATE 5,7:PRINT CHR$(244)
1180 LOCATE 17,11:PRINT CHR$(246)
1190 LOCATE 17,7:PRINT CHR$(246)
1200 MOVE 88,240:DRAW 88,240,11
1210 TAG: MOVE 80,246:PRINT CHR$(245);
1220 MOVE 88,226:DRAW 88,226,5
1230 MOVE 80,232:PRINT CHR$(245);
1240 MOVE 88,212:DRAW 88,212,9
1250 MOVE 80,218:PRINT CHR$(245);: TAGOFF
1260 RETURN
```

The DMP1 is more than an ordinary printer. It costs less than most - £199 retail, £169 to club members - and yet offers more. It's fast, inexpensive and versatile. It prints 80 columns - that means a full 80 characters across the page -and prints characters using a 5 x 7 dot matrix.

Using ordinary printer paper (expensive electrostatic paper is not needed), it prints characters using a special 'impact' technique. The DMP1 printer takes standard perforated fan-fold paper and can print 'dot graphics' for screen dumps as well as high quality character printouts.

How printers work

The first computer printers worked just like Telex printers. Codes for each character were sent to the printer, and these codes determined the sequence of printing bars popping up to hit the printing ribbon against the paper. Things have advanced from those early printers to create products such as the DMP1. Modern printers use a number of techniques that boil down into two basic printing methods - impact printing and non-impact printing.

Non impact printers (the DMP1 is not one of them) usually involve the electrostatic transfer of printing marks from the print head to specially treated paper. The marks are, in effect, burnt onto the paper. Another non-impact technique sometimes used in high speed, expensive printers is the so-called 'ink-jet' technique. This method involves squirting a jet of ink droplets at the paper that dry on contact. The method is silent and reasonably fast but very expensive.

An even faster - and yet more expensive - technique utilises laser printing. Laser printers work something like photocopiers. A high power laser beam is scanned onto a roller and causes it to become electrostatically charged. The roller then picks up ink powder and transfers it to the paper - all at very high speed. Laser printers can print several thousand lines of output a minute, but the equipment is cumbersome, expensive and beyond the means of the home computer user.

Impact printers

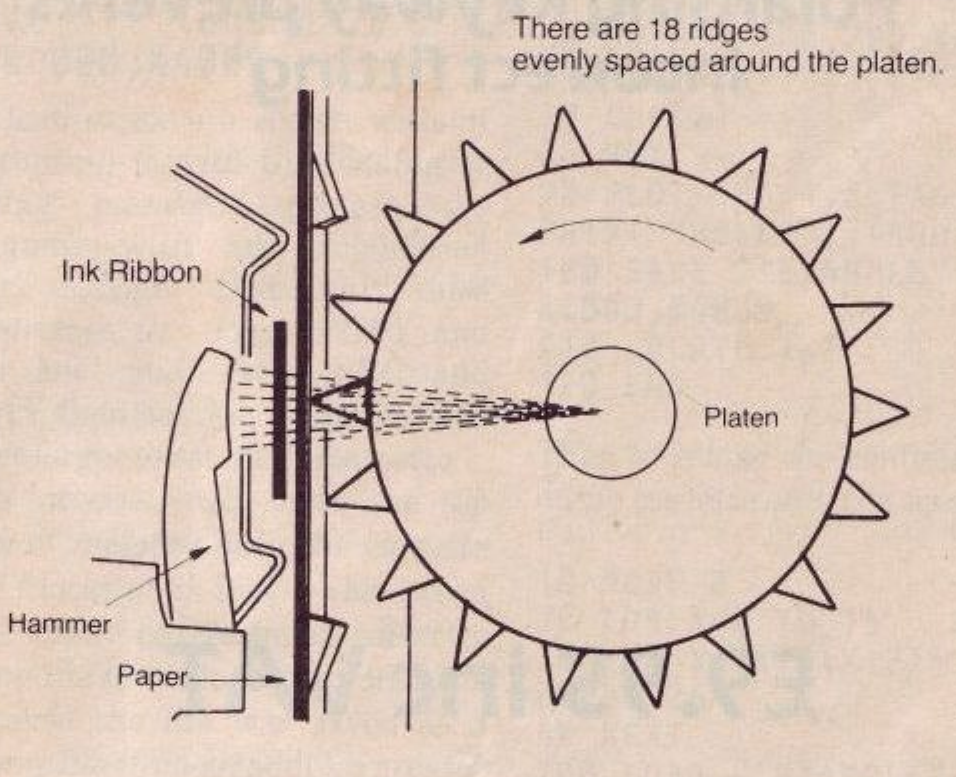
The slowest printing technique of all involves the use of so-called 'daisy wheel' printers. These printers use a disc constructed like a flower, where each 'petal' has a reversed letter or digit moulded into the end. When a code corresponding to a letter or digit comes from the computer to the printer, the 'daisy wheel' is rotated until the appropriate character is in position against the printing ribbon. An impulse is then sent to a hammer that hits the 'petal' against the ribbon to print the character on the paper.

Dot matrix printers

By far the vast majority of computer printers use a more advanced technique whereby one or more small pins is 'fired' against the ribbon to print a series of dots. Such printers are called 'dot matrix printers'. Special software built into the Amstrad DMP1 printer interprets the character codes that are sent from the computer and ensures that each dot is printed on the paper in exactly the right position to create a single character.

Each character, then, is made up of a matrix of tiny dots - up to five dots across and seven dots down in the case of the DMP1. The illustration below shows how this is done.

Some dot printers, including the DMP1, incorporate special software routines that allow every dot to be printed under the direct



Printing Mechanism

NB Although producing dot matrix print-style, the DMP-1 uses the patented uni-hammer technique developed by Seikosha

control of the user. This allows user-defined graphic symbols to be printed as well as letters and digits, and makes printing an exact replica of the images on the computer screen possible - a process called screen dumping. The DMP1 allows 60 dots to be printed per inch in the graphics mode.

The Final Word

Computers can be great fun even without a printer. But to get the most from them, you should seriously consider investing in a printer too. Printers are an essential accessory if you want to develop your own programs, almost indispensible if you want to do financial planning using a spread-sheet program, and word processing without a printer would be like typing without paper.

Your CPC464 is provided with an industry standard 'Centronics' style printer interface, so almost any commercially available printer would work with it, but, like your computer, the Amstrad DMP1 printer would work with it, but, like your computer, the Amstrad DMP1 printer gives more performance for less outlay, and should come right at the top of your list.

Whether you want to print out program listings to check for bugs and errors, financial projections, letters to clients or anything else that needs a permanent record of what's going on in your computer, you need a printer!

■CPC464 USER

Page 12 *In the case of a vendor (which may be a private individual or company):*

Program Title
Intended applications
Name and address of vendor
The price
Our reference number

In the case of a program wanted:

Intended applications
Name and address
Our reference number

We require all submissions from vendors of software for this directory to contain the following:

1 Tape of the program (as proof it exists)
2 A 25 word description of its purpose
3 The name and address for correspondance
4 Price

NB AMSOFT offers this listing as a free service. We cannot be responsible for any of the commercial arrangements between the vendor and purchaser, and all prospective purchasers should remember the old saying 'caveat emptor' - or buyer beware.

We will make no comment whatsoever on the program in the directory. The entry simply advises that such a program exists, and provides the means by which users may get together.

AMSOFT reserves the right to decline to publish any entry without notice or reason. Correspondance can only be entered into where a stamped self-addressed envelope is supplied.

Programs submitted for the prize competition may be entered subsequently for the directory -but not until after the competition has been concluded!

Q. In the dumps....

Dear Sir,
I recently ordered a CPC464 from you, and was very pleased when it arrived after only two weeks.

After reading the manual, there are a couple of things I would like to clear up.

I would like to know how to dump screen data on to a cassette (chapter 2.6.4).

Also, in Appendix IV it says that a composite video signal is present at the monitor socket, and I would like to know which pins to use.

SR, Hull.

A. SAVEd by the CPC464

Screen dumps are simply performed by using a binary save command. Because of the screen layout, it is not easy to dump part of the screen, but only a whole screen. The command to use is:

```
SAVE "<filename>",
b,&C000,&4000
```

To load or save a screen without corrupting it with the loading or saving messages, preface the filename with an exclamation mark ! Note that this also suppresses the 'Press PLAY and any key' and 'Press REC and PLAY then any key' messages - so don't forget to set up the cassette. This process then saves the top 16K of memory onto the cassette in 8 blocks of 2k bytes. The screen can then be reloaded directly using the usual loading procedure. Because the file was saved as a binary file from an address greater than HIMEM, it will be loaded back to exactly the same address that it was saved from.

Note that during the reloading process, data is first transferred to the cassette buffer, but once the header information has been processed and the file been recognised as being of the type described, subsequent blocks are loaded directly to the screen RAM. You will see exactly how the screen RAM of the CPC464 is addressed as the bytes are loaded consecutively.

Note that the information concerning screen mode is not saved during this process - the screen should be in the correct mode before you reload. Also note that the memory contains the details of pixel positions and inks -not colours, so it will be necessary to set the inks to the colours used when the screen was saved.

Further, note that the CPC464 performs vertical screen movement by adjusting the relationship between memory and screen position. If a reloaded screen is to appear in the same place, then it is important to have the same memory to screen mapping. The simplest mapping to choose is the mapping just after a mode change.

```
10 MODE 0
20 FOR X=0 TO 15
30 INK X,(X+((X\2)*13)+
8)MOD 26
40 NEXT
50 FOR X=0 TO 15
60 ORIGIN 320,200,X*16,
638-X*16,X*12,398-X*12
70 CLG(X)
80 NEXT
90 PLOT -80,4,0:TAG:
PRINT "HELLO";:TAGOFF
100 SAVE "!EXAMPLE",B,
&C000,&4000
110 LOCATE 1,1
120 END
```

Then to reload the contents of the dump cassette onto the screen:-

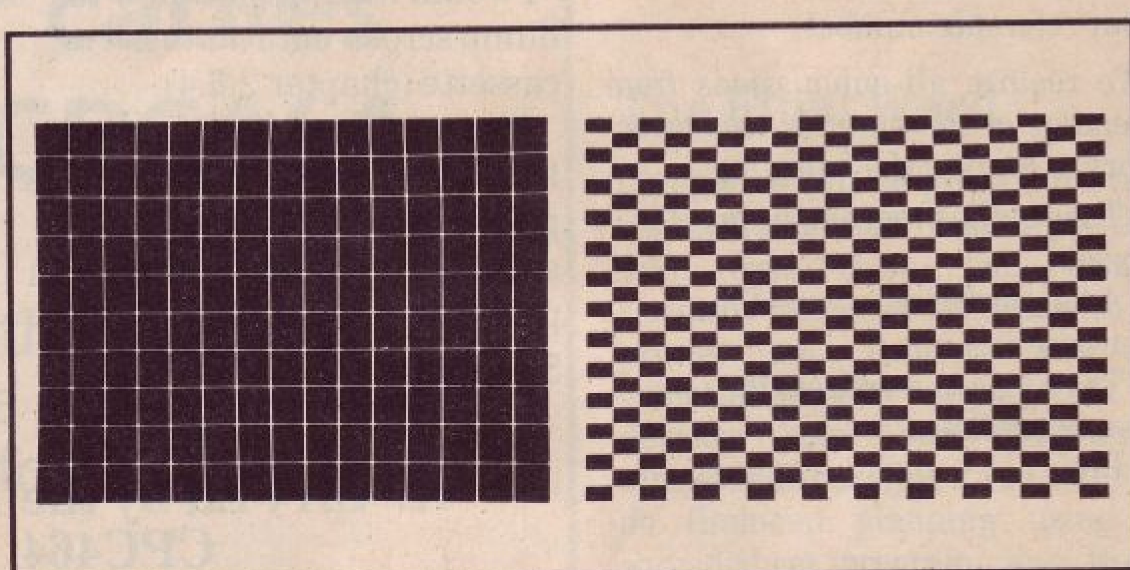
```
10 MODE 0
20 FOR X=0 TO 15
30 INK X,(X+((X\2)*13)+
8)MOD 26
40 NEXT
100 LOAD "!EXAMPLE"
110 LOCATE 1,1
120 GOTO 120
```

And finally, Pin 6 of the video connector supplies the composite (mono) video signal, containing both luminance and sync.

Continued on page 30

Programs

In this issue, we present three very different programs that have one thing in common: they all illustrate the tremendous power of the CPC464 and show just how much can be be achieved with really quite short programs written in BASIC.



In this, the first issue of the User Club magazine, we start a regular feature with a double aim: to present programs for you to type in and try out on your CPC464, and in so doing, to explain some of the programming techniques used. The level of explanation will range from the very simple (to help the beginning programmer) to more detailed descriptions of some of the advanced techniques used by experienced programmers. In this way, we hope that you will gain insights that the text books often miss out, and at the same time have fun using your new computer in a constructive way.

We have a three dimensional drawing program from David Robinson of Saxon Computing that is quite unusually compact for such a sophisticated application. Another graphics program exploits the window capabilities of the CPC464 and shows how easy it all can be.

Finally, we have a short program for producing colour bar charts and other patterns for examining the performance of TVs and monitors. Don't be surprised to find that the performance of the dedicated CTM640 monitor is superior to the modulator/TV combination.

WINDOWS

This program demonstrates the ease with which 'windows' (separately addressable portions of the screen) and be set up and written to (lines 160,230 and 240) After the windows have been set up and had the relevant captions and axes drawn in, initial values for the variables used in the subroutines are established in lines 350 and 360.

The code from lines 410 to 430 are much like the ordinary subroutine calls except that the EVERY statement is used. EVERY allows subroutines to be called at regular, specified intervals. The interval is specified by the first argument, and the number of the timer by the second. The EVERY statement calls the subroutine associated with it after the specified delay has elapsed. Each delay interval is 1/50th of a second.

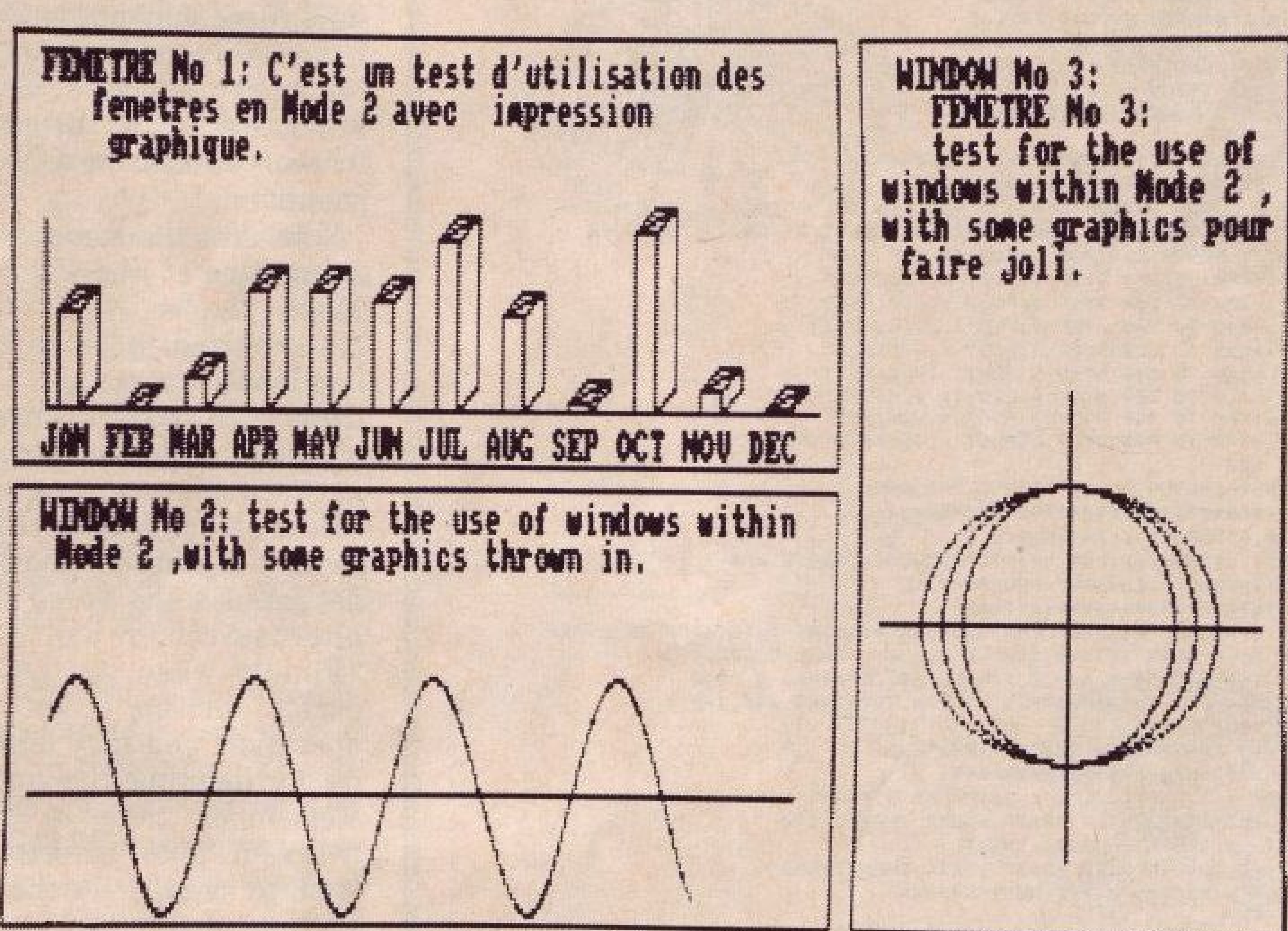
Line 420, therefore, calls the ellipse subroutine every 0.1 seconds; the associated timer is number 1. For a complete description of the EVERY command, refer to chapter 10, page 2 of the user instruction book. Note also the use of DI (Disable Interrupts) and EI (Enable Interrupts) in the ellipse and bargraph routines. Try REMing these out to see the effect. Another important line is

```
440 GOTO 440
```

The program will not work if this is taken out (by REM). Can you see why?

The three subroutines are all fairly straightforward and can easily be adapted for use in other programs. Instead of using the RND (random) function (line 770) to set the height of the bars, the value of T could be supplied from other parts of the your program to make a very easy and efficient bar graph generator.

```
10 REM ****************************************
20 REM ****************************************
30 REM ****   USING WINDOWS AND GRAPHICS    ***
40 REM ****************************************
50 REM ****************************************
60 REM
70 MODE 2
80 CLS
90 BORDER 26
100 INK 0,20
110 INK 1,1
120 REM
130 REM ****************************************
140 REM ***         CREATE WINDOWS           ***
150 REM ****************************************
160 WINDOW #1,3,52,2,12
170 PLOT 10,10
180 DRAW 419,10,1
190 DRAW 419,196,1:DRAW 10,196:DRAW 10,10
200 PLOT 10,206:DRAW 419,206:DRAW 419,390:DRAW 10,390:DRAW 1
0,206
210 PLOT 430,10:DRAW 630,10:DRAW 630,390:DRAW 430,390:DRAW 4
30,10
220 PLOT 410,230:DRAW 26,230:DRAW 26,310
230 WINDOW #2,3,52,14,24
240 WINDOW #3,56,78,2,24
250 PRINT #1," WINDOW No 1: Using windows and graphics in
                Mode 2.
                                                        ***RA
NDOM BAR-GRAPHS*** "
260 PRINT #2, " WINDOW No 2:    ***PLOTTING A SINE WAVE***  "
270 PRINT #3, " WINDOW No 3:
                                 ***ELLIPSES***"
280 PLOT 20,65:DRAW 400,65
290 PLOT 440,140:DRAW 620,140:PLOT 530,40:DRAW 530,240
300 LOCATE 4,12:PRINT "JAN FEB MAR APR MAY JUN JUL AUG SEP O
CT NOV DEC"
310 REM
320 REM ****************************************
330 REM ***       SET UP VARIABLES           ***
340 REM ****************************************
350 A=0:N=-680
360 M=1
370 REM
380 REM ****************************************
390 REM ***       SUBROUTINE CALLS           ***
400 REM ****************************************
410 EVERY 5,1 GOSUB 600:     ' Ellipse
420 EVERY 3,2 GOSUB 490:     ' Sinewave
430 EVERY 130 ,3 GOSUB 720: ' Bar-graph
440 GOTO 440
450 REM
460 REM ****************************************
470 REM ***      SINEWAVE SUBROUTINE         ***
480 REM ****************************************
490 DEG:ORIGIN 200,65
500 Y=SIN(N)
510 PLOT N*0.25 ,50*Y
520 N=N+2
530 IF N=680 THEN GOTO 880
540 LET LP = LP + 1
550 RETURN
560 REM
570 REM ****************************************
580 REM ***       ELLIPSE SUBROUTINE         ***
590 REM ****************************************
600 DI:      ' Disable interrupt
610 DEG:ORIGIN 530,140
620 PLOT 50*COS(A),60*SIN(A)
630 PLOT 60*COS(A),60*SIN(A)
640 PLOT 70*COS(A),60*SIN(A)
650 A=A+2
660 EI:      ' Enable interrupt
670 RETURN
680 REM
690 REM ****************************************
700 REM ***   RANDOM BARGRAPH SUBROUTINE    ***
710 REM ****************************************
720 DI
730 IF M>12 THEN RETURN
740 ORIGIN 0,0
750 X=M*32
760 Y=230
770 T=INT(RND(1)*80)
780 PLOT X,Y:DRAW X+10,Y:DRAW X+20,Y+10
790 DRAW X+20,Y+T+10:DRAW X+10,Y+T+10
800 DRAW X,Y+T: DRAW X,Y:DRAW X+10,Y
810 DRAW X+10,Y+T
820 DRAW X+20,Y+T+10: DRAW X,Y+T
830 DRAW X+10,Y+T
840 M=M+1
850 EI
860 RETURN
870 REM End of program routine
880 PRINT CHR$(7):                ' Beep when finished
890 IF INKEY$ = "" THEN 890:    ' Do nothing until key is pre
ssed
900 CLS: END
```



French version of above listing.

```
10 REM **************** TEST  PATTERNS ****************
20 REM **************** by Ivor Spital ****************
30 REM ***********************************************
40 REM *** (c)AMSTRAD CONSUMER ELECTRONICS PLC.1984 ***
5Ø 'ON BREAK GOSUB 82Ø
6Ø SPEED INK 2Ø,2Ø
7Ø MODE 1:BORDER 1:INK Ø,1:INK 1,24:INK 2,1,24:PAPER Ø:PEN 1
8Ø PRINT TAB(9)"AMSTRAD CPC-464 SYSTEM"
9Ø PRINT:PRINT TAB(6)"Computer/Monitor Test Pattern"
1ØØ LOCATE 1Ø,1Ø:PRINT"1  Crosshatch"
11Ø LOCATE 1Ø,12:PRINT"2  Low Frequency"
12Ø LOCATE 1Ø,14:PRINT"3  Colour bar"
13Ø LOCATE 1Ø,16:PRINT"4  Bandwidth/Geometry"
14Ø LOCATE 1Ø,18:PRINT"5  Colour Purity/Sound"
15Ø LOCATE 1Ø,2Ø:PRINT"6  RGB Alignment"
16Ø LOCATE 16,25:PEN 2:PRINT"Select?";:PEN 1
17Ø a$=INKEY$:IF a$="" THEN 17Ø
18Ø IF a$="1" THEN 25Ø
19Ø IF a$="2" THEN 29Ø
2ØØ IF a$="3" THEN 37Ø
21Ø IF a$="4" THEN 5ØØ
22Ø IF a$="5" THEN 64Ø
23Ø IF a$="6" THEN 68Ø
24Ø GOTO 17Ø
25Ø REM *** CROSSHATCH PROGRAM
26Ø REM **********************
27Ø MODE Ø:GOSUB 75Ø
28Ø GOTO 28Ø
29Ø REM *** LOW FREQUENCY PROGRAM
3ØØ REM *************************
31Ø MODE Ø:BORDER Ø:INK Ø,Ø:INK 1,26:PAPER Ø:PEN 1
32Ø FOR y=1 TO 25 STEP 2:FOR x=1 TO 2Ø STEP 2
33Ø LOCATE x,y:PRINT CHR$(143);:NEXT:NEXT
34Ø FOR y=2 TO 25 STEP 2:FOR x=2 TO 2Ø STEP 2
35Ø LOCATE x,y:PRINT CHR$(143);:NEXT:NEXT
36Ø GOTO 36Ø
37Ø REM *** COLOUR BAR PROGRAM
38Ø REM **********************
39Ø MODE Ø:BORDER Ø
4ØØ INK Ø,Ø:INK 1,25:INK 2,23:INK 3,21:INK 4,17:INK 5,6:INK 6,2:INK 7,26
41Ø PAPER Ø:CLS
42Ø PAPER 1:WINDOW 2,4,1,18:CLS
43Ø PAPER 2:WINDOW 5,7,1,18:CLS
44Ø PAPER 3:WINDOW 8,1Ø,1,18:CLS
45Ø PAPER 4:WINDOW 11,13,1,18:CLS
46Ø PAPER 5:WINDOW 14,16,1,18:CLS
47Ø PAPER 6:WINDOW 17,19,1,18:CLS
48Ø PAPER 7:WINDOW 2,19,19,25:CLS
49Ø GOTO 49Ø
5ØØ REM *** BANDWIDTH/GEOMETRY PROGRAM
51Ø REM ******************************
52Ø MODE 2:GOSUB 75Ø:ORIGIN Ø,Ø:FOR c=Ø TO 36Ø:DEG:PLOT 32Ø,2ØØ
53Ø PLOT 32Ø+19Ø*COS(c),2ØØ+19Ø*SIN(c):NEXT:WINDOW 21,6Ø,8,18
54Ø PAPER 1:PEN Ø:CLS:FOR y=287 TO 11Ø STEP -1
55Ø FOR x=16Ø TO 2ØØ STEP 9:PLOT x,y,Ø:NEXT
56Ø FOR x=2ØØ TO 24Ø STEP 8:PLOT x,y,Ø:NEXT
57Ø FOR x=24Ø TO 28Ø STEP 7:PLOT x,y,Ø:NEXT
58Ø FOR x=28Ø TO 32Ø STEP 6:PLOT x,y,Ø:NEXT
59Ø FOR x=32Ø TO 36Ø STEP 5:PLOT x,y,Ø:NEXT
6ØØ FOR x=36Ø TO 4ØØ STEP 4:PLOT x,y,Ø:NEXT
61Ø FOR x=4ØØ TO 44Ø STEP 3:PLOT x,y,Ø:NEXT
62Ø FOR x=44Ø TO 48Ø STEP 2:PLOT x,y,Ø:NEXT:NEXT
63Ø GOTO 63Ø
64Ø REM *** COLOUR PURITY/SOUND PROGRAM
65Ø REM ******************************
66Ø INK Ø,6:BORDER 6:PAPER Ø:CLS
67Ø SOUND 1,2ØØ,4Ø,7:FOR t=1 TO 1ØØØ:NEXT:GOTO 67Ø
68Ø REM *** RGB ALIGNMENT PROGRAM
69Ø REM ************************
7ØØ MODE 1:BORDER 1:SPEED INK 5,5:INK Ø,1:INK 1,18,6:PAPER Ø:PEN 1
71Ø MOVE 2Ø,Ø:DRAW 2Ø,398,1:MOVE Ø,378:DRAW 638,378,1
72Ø MOVE 618,398:DRAW 618,Ø,1:MOVE 638,2Ø:DRAW Ø,2Ø,1
73Ø MOVE 32Ø,Ø:DRAW 32Ø,398,1:MOVE Ø,199:DRAW 64Ø,199,1
74Ø GOTO 74Ø
75Ø REM *** CROSSHATCH SUBROUTINE
76Ø REM ************************
77Ø BORDER 26:INK Ø,Ø:INK 1,26:PAPER Ø:PEN 1
78Ø FOR y=Ø TO 4ØØ STEP 4Ø:IF y>39Ø THEN y=398
79Ø ORIGIN Ø,y:DRAW 638,Ø,1:NEXT
8ØØ FOR x=Ø TO 64Ø STEP 4Ø:IF x>63Ø THEN x=638
81Ø ORIGIN x,Ø:DRAW Ø,398:NEXT:RETURN
82Ø RUN
```

TEST PATTERNS

Ivor Spital - AMSTRAD

There are six different display options, similar to those generated by TV test pattern generators used for aligning colour TV equipment. Lines 170 to 240 use INKEY$ to read a number from 1 to 6 from the keyboard (to select the option) and then branch to the appropriate routine.

After the routine has been selected, the program will 'hang' there as a result of the n GOTO n command which terminates each routine. The chosen pattern will then stay on the screen until the [ESC] key has been pressed twice. Note, however that we have REMed out line 50 so that you can break the program using the [ESC] key. (The alternative form of REM, an apostrophe, is used in line 50.). If line 50 was not disabled by REM, you could not then break from the program to check the listing or change it, unless BASIC tripped over an error -and there may not be one if you have been conscientious. Always remember to save yourself a breakable copy of programs where you use the ON BREAK command.

Note that the same crosshatch subroutine at line 750 is used in Mode 0 for the crosshatch pattern, and in Mode 2 for the bandwidth/geometry pattern, demonstrating the compatibility of the graphics coordinates between different screen modes.

Note also the SPEED INK statement in lines 60 and 700. This determines the speed at which alternate colours will flash.

Finally when you have tested each routine for correct operation, you may unREM line 50 by deleting the apostrophe. Remember then to save the program onto cassette if you wish to keep, or further develop it, as once you RUN the program, the only way you will be able to break from it, will be by resetting the computer.

CAD for beginners -Part 1

David Robinson of Saxon Computing shows how the CPC464 can produce 3D drawings from simple 2D sections.

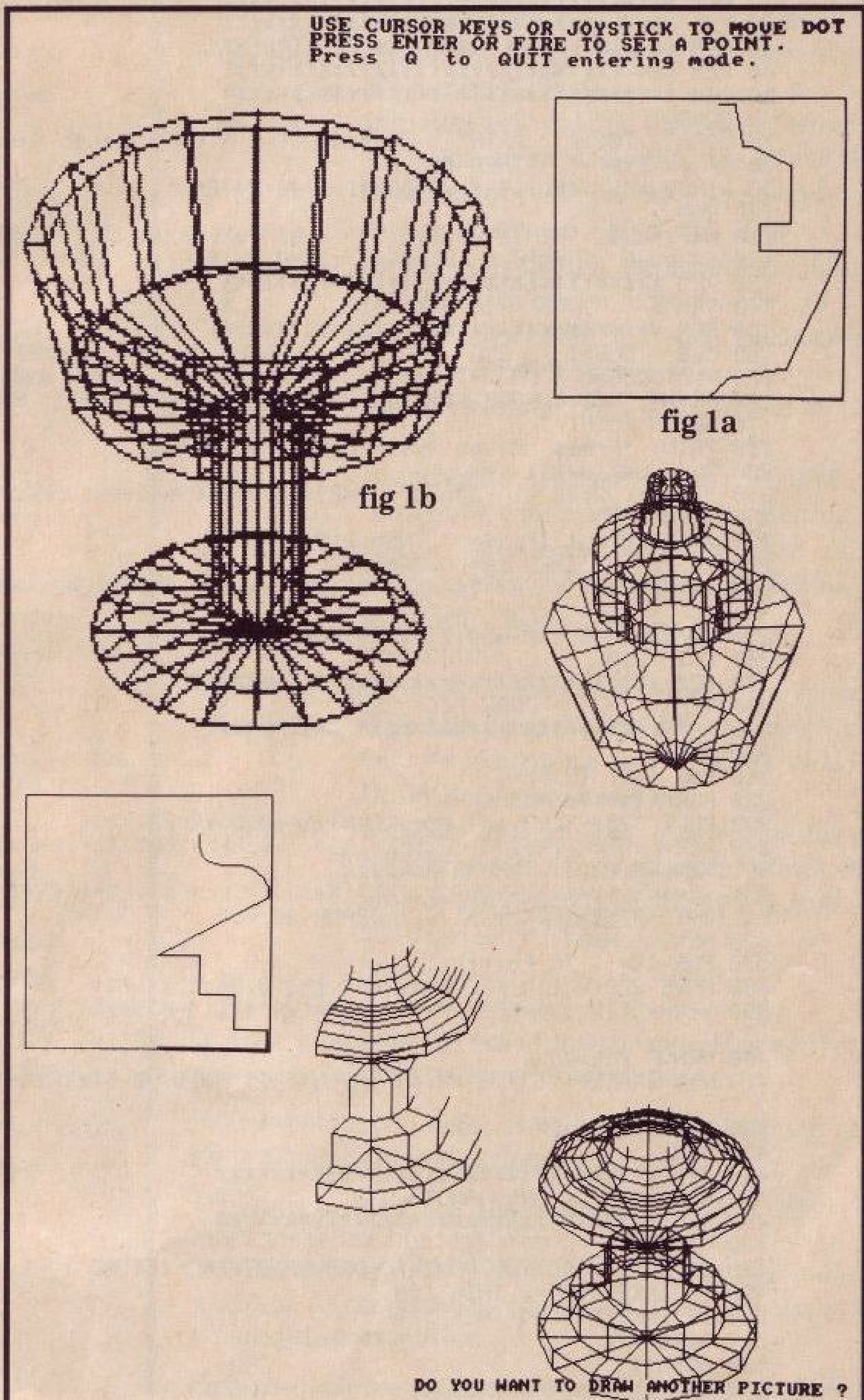


fig 1a

fig 1b

Around the end of May I was asked if I'd present a graphics program which I'd evolved during my acquaintance with the CPC464 in the launch issue of AMSTRAD USER. The ideas began to flow, and then grew into a regular column for the magazine on the subject of computer graphics, and so now I have a splendid excuse, at least once a month, to get away from writing business software!

Some graphics applications require so much computing that even the Locomotive BASIC inside your 464 (just about the fastest standard 8 bit BASIC around) can seem a bit slow. Where a program could do with some extra speed I will list the program both in BASIC and PASCAL for those of you with the HiSoft PASCAL 4T compiler.

Every program is accompanied by a full commentary and a list of suggested improvements and extensions for you to try out yourself. Your CPC464 has some very useful graphics features not found on any other micro under £500. These allow us to use techniques more associated with mainframe graphics computers than most micros.

Three other important features are:-

1) a palette of 27 colours to choose from.
2) switchable screen resolution - but using a common co-ordinate base.
3) VERY IMPORTANT - the colour resolution is the same as the pixel resolution.

A 3D wire frame plotter

This program of 3D drawings translates a two dimensional cross section into a full 3D shape. To get an idea of how it works, look at Figure 1a which shows the right half of the cross section of a wine glass. The left half is, of course, a simple mirror image, so the program does not require us to draw both halves. Figure 1b shows the 3D projection drawn by the computer from the cross section.

■ CPC464 USER

```
10 REM **********************************
20 REM **********************************
30 REM ** WIRE FRAME GRAPHICS PLOTTER **
40 REM **********************************
50 REM **********************************
60 MODE 1
70 LOCATE 10,12 : PRINT "   Press  S P A C E  B A R  to start"
80 IF INKEY$<>" " THEN 80
90 WINDOW#0,1,40,1,4 : WINDOW#1,1,40,24,24
100 RAD
110 maxi%=20 : DIM x%(maxi%,50) : DIM y%(maxi%,50) : DIM z%(maxi%,50)
120 dx%=320 : dy%=0 : i0%=26 : i1%=0 : i2%=13 : i3%=6 : Point%=0
130 REM **********************************
140 REM *       GET PICTURE DATA         *
150 REM **********************************
160 CLS : CLS#1 :  CLG : BORDER i2% :INK 0,i0% :  PAPER 0 : INK 1,i1% : PEN 1
170 PLOT 156,0 : DRAW 156,304 : DRAW 494,304 : DRAW 494,0
180 PRINT "USE CURSOR KEYS OR JOYSTICK TO MOVE DOT" : PRINT "PRESS ENTER OR FIRE
 TO SET A POINT."
190 PRINT "Press  Q  to  QUIT entering mode."
200 PLOT dx%,dy%,1
210 z%=-1*(JOY(0)=1)-2*(JOY(0)=2)-3*(JOY(0)=4)-4*(JOY(0)=8)-5*(JOY(0)=16) : IF z
%>0 THEN 250
220 z$=INKEY$ : IF z$="" THEN 210
230 IF z$="Q" THEN z$=LOWER$(z$)
240 x%=ASC(z$) : z%=-1*(x%=240)-2*(x%=241)-3*(x%=242)-4*(x%=243)-5*(x%=13)-6*(x%
=113)
250 ON z% GOTO 620,690,760,830,900,1020
260 GOTO 210
270 REM **********************************
280 REM *         DRAW PICTURE           *
290 REM **********************************
300 CLG
310 FOR l%=1 TO f%-1
320 FOR k%=1 TO Point%-1
330 PLOT 320+x%(k%,l%),80+y%(k%,l%)-0.5*z%(k%,l%)
340 xc%=(x%(k%,l%+1))-(x%(k%,l%)) : yc%=(y%(k%,l%+1)-0.5*z%(k%,l%+1))-(y%(k%,l%)
-0.5*z%(k%,l%)) : DRAWR xc%,yc%
350 xc=(x%(k%+1,l%+1))-(x%(k%,l%+1)) : yc=(y%(k%+1,l%+1)-0.5*z%(k%+1,l%+1))-(y%(
k%,l%+1)-0.5*z%(k%,l%+1)) : DRAWR xc,yc
360 NEXT : NEXT
370 FOR l%= 1 TO Point%-1
380 MOVE 320+x%(l%,f%),80+y%(l%,f%)-0.5*z%(l%,f%)
390 xc%=x%(l%,1)-x%(l%,f%) : yc%=(y%(l%,1)-0.5*z%(l%,1))-(y%(l%,f%)-0.5*z%(l%,f%
))
400 DRAWR xc%,yc%
410 xc%=x%(l%+1,1)-x%(l%,1) : yc%=y%(l%+1,1)-0.5*z%(l%+1,1)-y%(l%,1)-0.5*z%(l%,1
)
420 DRAWR xc%,yc%
430 NEXT
440 REM **********************************
450 REM *       DRAW ANOTHER ?           *
460 REM **********************************
470 CLS#1
480 PRINT#1, "DO YOU WANT TO DRAW ANOTHER PICTURE ?"
490 IF INKEY(43)=0 THEN 120
500 IF INKEY(46)=0 THEN 520
510 GOTO 490.
520 MODE 1
530 END
```

```
540 REM ********************************
550 REM ********************************
560 REM **   END OF MAIN PROGRAM      **
570 REM ********************************
580 REM ********************************
590 REM
600 REM
610 REM
620 REM ********************************
630 REM *           MOVE UP            *
640 REM ********************************
650 yp%=dy% : dy%=dy%+2
660 IF dy%>300 THEN dy%=300
670 PLOT dx%,yp%,0
680 GOTO 200
690 REM ********************************
700 REM *           MOVE DOWN          *
710 REM ********************************
720 yp%=dy% : dy%=dy%-2
730 IF dy%<0 THEN dy%=0
740 PLOT dx%,yp%,0
750 GOTO 200
760 REM ********************************
770 REM *           MOVE LEFT          *
780 REM ********************************
790 px%=dx% : dx%=dx%-2
800 IF dx%<160 THEN dx%=160
810 PLOT px%,dy%,0
820 GOTO 200
830 REM ********************************
840 REM *           MOVE RIGHT         *
850 REM ********************************
860 px%=dx% : dx%=dx%+2
870 IF dx%>490 THEN dx%=490
880 PLOT px%,dy%,0
890 GOTO 200
900 REM ********************************
910 REM *           SET POINT          *
920 REM ********************************
930 Point%=Point%+1
940 x%(Point%,1)=dx%-320
950 y%(Point%,1)=dy%
960 z%(Point%,1)=0
970 IF Point%>1 THEN PLOT x%(Point%-1,1)+320,
    y%(Point%-1,1) : DRAW x%(Point%,1)+
320,y%(Point%,1),1
980 IF Point%=maxi% THEN 1020
990 IF JOY(0)<>0 THEN 990
1000 IF INKEY$<>"" THEN 1000
1010 GOTO 200
1020 REM ********************************
1030 REM *           END INPUT          *
1040 REM ********************************
1050 CLS
1060 INPUT "ENTER NUMBER OF FACES ";f%
1070 IF f%>50 OR f%<4 THEN 1050
1080 CLS : PRINT "PLEASE WAIT - COMPUTING
     POINTS MATRIX"
1090 sn=SIN(PI/(f%/2)) : cs=COS(PI/(f%/2))
1100 FOR l%=2 TO f%
1110 FOR k%=1 TO Point%
1120 x%(k%,l%)=x%(k%,l%-1)*cs-z%(k%,l%-1)*sn
1130 y%(k%,l%)=y%(k%,l%-1)
1140 z%(k%,l%)=z%(k%,l%-1)*cs+x%(k%,l%-1)*sn
1150 NEXT
1160 NEXT
1170 GOTO 270
```

Significant line numbers are as follows:

Lines

60	Sets the screen at 320 x 200 resolution.
70-90	Sets up two text windows so that we can keep messages separate from the graphics.
100-110	Initialise 3 arrays storing the co-ordinates of points in 3D space.
120	Initialise dx% and dy% - the co-ordinates of your 'dot cursor' -note that the LOGICAL co-ordinates are always expressed as though there are 640 x 400 dots on the screen even though the actual resolution changes with screen mode. This saves you having to convert all your calculations every time you change screen mode.
160-260	Collect data and draw cross section on the screen. Branches to separate routine at 1600 to complete computation of data for plotting.
300-430	Draws successive cross sections and connects each one to the previous section so as to form a 3D picture.
470-530	End routine.
650-890	Moves your 'dot cursor' around the screen,
930-1010	Puts data for each point into the arrays x% and y% describing the 2D cross section. Draws line to previous point.
1060-1140	Accepts data on the amount of detail (FACES) in the final picture. Rotates the 2D cross section data in 3D space in steps of 360 degrees/FACES.

A point to note is the use of integer variables wherever possible to improve the speed of computation.

You could try to make some improvements to the program along the following lines -

1) Draw a line along the centre of the input screen to mark the centre line for the cross section.
2) The 'dot cursor' will erase any previously drawn lines whilst on its travels about the screen. See if you can write a routine to prevent this.

Save the program on tape as we shall be using it again in later issues.

Bustout!

by Alexander Martin

Type in your first game

One of the most important features of the CPC464 is its ability to do a lot with only a minimum of programming effort. The programs in this issue demonstrate this facet of the system, and this version of the well known 'breakout' style of game manages to fit all the essential features into a single page listing.

There's even a comprehensive range of sounds to support the visual activity.

The cursor keys or joystick control the bat that runs along the bottom of the screen. Stop the 'ball' (it's actually a square, but you could chose another character. Line 300 sets up the bat with character 233.

```
1Ø BORDER 1:INK Ø,1:INK 1,26:INK 2,24:INK 3,6
2Ø SPEED KEY 15,2
3Ø ENV 1,1,18,Ø,11,Ø,1Ø:ENT 1,1Ø,2,2
4Ø ENV 3,1,Ø,16,5,-3,2
5Ø ENV 2,5,3,3,1,-21,22,9,-3,2:ENT -2,1Ø,2,2,5,-7,1,2,11,3,2
,-4,8
6Ø MODE 1
7Ø MOVE 3Ø,16:DRAWR Ø,4ØØ,1:MOVE 61Ø,16:DRAWR Ø,4ØØ,1
8Ø PEN 3:LOCATE 3,1:PRINT STRING$(36,143):PEN 2:LOCATE 3,2:P
RINT STRING$(36,143):PEN 1:FOR r=5 TO 6:LOCATE 3,r:PRINT STR
ING$(36,143):NEXT r
9Ø bx=9
1ØØ lives=5:score=Ø
11Ø PEN 1:GOSUB 42Ø
12Ø IF INKEY$<>" "THEN 12Ø
13Ø GOTO 16Ø:REM start the game
14Ø LOCATE bx,24:PRINT"  ";STRING$(4,131);"  "
15Ø RETURN
16Ø xa=1:ya=1:IF INT(RND*2)=1 THEN xa=-xa:REM for each life
start here
17Ø PEN 1:GOSUB 14Ø
18Ø ORIGIN Ø,4ØØ
19Ø x=bx+4:y=11:x1=x:y1=y
2ØØ x1=x+xa:y1=y+ya:REM start of main loop
21Ø IF x1=3 OR x1=38 THEN xa=-xa
22Ø GOSUB 36Ø
23Ø IF y1=24 AND x1>bx+1 AND x1<bx+6 THEN ya=-ya:y1=y1-2:SOU
ND 13Ø,44,8,7,1,1:a=((x>bx+5)OR(x<bx+2)):IF a=-1 THEN xa=xa*
a:x1=x1+xa:y1=y1+1
24Ø IF y1=25 THEN LOCATE x,y:PRINT" ":GOTO 33Ø
25Ø GOSUB 14Ø
26Ø t=TEST((16*x1)-1,-(16*y1)-1):IF t<>Ø THEN ya=-ya:xz=x1:y
z=y1:y1=y1+ya:GOSUB 39Ø:IF t=2 THEN score=score+1Ø:GOSUB 42Ø
27Ø IF t=3 THEN score=score+2Ø:GOSUB 42Ø
28Ø IF t=1 THEN score=score+5:GOSUB 42Ø
29Ø IF y1=1 THEN ya=1
3ØØ LOCATE x,y:PRINT " ":LOCATE x1,y1:PRINT CHR$(233):x=x1:y
=y1
31Ø IF y=1 OR x=3 OR x=38 THEN SOUND 129,78,8,7,1,1
32Ø GOTO 2ØØ:REM finish of main loop
33Ø lives=lives-1:SOUND 132,19,46,12,2,2:IF lives=Ø THEN GOT
O 4ØØ
34Ø GOSUB 42Ø
35Ø GOTO 16Ø
36Ø IF (INKEY(8)=Ø OR INKEY(74)=Ø)AND bx>2 THEN bx=bx-2:RETU
RN
37Ø IF(INKEY(1)=Ø OR INKEY(75)=Ø)AND bx<32 THEN bx=bx+2:RETU
RN
38Ø RETURN
39Ø LOCATE xz,yz:PRINT " ":RETURN
4ØØ IF score>=hiscore THEN hiscore=score
41Ø score=Ø:lives=5:GOSUB 42Ø:GOTO 6Ø
42Ø SOUND 13Ø,Ø,2Ø,13,3,Ø,31:LOCATE 1,25:PRINT"HISCORE";hisc
ore;:LOCATE 16,25:PRINT"SCORE";score:LOCATE 3Ø,25:PRINT"LIVE
S";lives:RETURN
```

Commentary

Lines

10-60	Set up the constants for the program. The sound tone and volume envelopes and mode.
70-130	Draws the new opening screen.
140-150	Draw the 'bat'.
160-190	Every time a life is lost (ie you miss the ball and it drops off the bottom of the display) this section generates a random entry for the next ball.
200-320	The main program loop. It 'Undraws' the ball (by overprinting the old position with a space), then draws the new position. The routine also spots the boundaries and produces an appropriate noise - and increments the score when the topline is eaten away.

Subroutines

330-350	This provides the 'end of a life' condition if you miss the ball in 240.
360-380	This routine, called from 220, scans the keyboard or joystick to move the bat.
390	Called from the midst of 260, this routine erases the block that has been hit by the ball by replacing it with a space character.
400-410	When the lives reach zero (line 330), this re-initialises the life count and score before restarting at line 60.
420	The score update routine, called from 260, 270 or 280, depending on which block has been knocked from the 'wall'.

CPC464 USER £2000 program competition!

Think what you could buy with a £2000 windfall ......

The objective couldn't be simpler: write the best program submitted to the CPC464 user review panel. Every other month £2000 will be awarded to the best submission.

HOW TO ENTER:

Think about your program, and map it out in a series of events or features. Write the program onto cassette based around these events and check that the program runs as intended. Once you are satisfied, send a copy of the cassette in a suitable envelope along with the following:

1) A brief summary of the program in 500 words or less.
2) The completed competition entry form, which must be signed.
3) Program listings, if available.
4) SAE if want your entry returned.

RULES

1 The winner(s) will be the person or group who submits the best program. It may be any piece of software that runs on the CPC464.

2 The name of the winner will be printed in the next issue of CPC464 USER.

3 All entries must arrive by September 31st 1984. Entries arriving after that date will be included in the next issue's competition.

4 All entries must INCLUDE a cassette copy of the program (plus loading instructions where necessary!), AND A SIGNED COMPETITION ENTRY FORM. A full listing would be helpful, but not essential. A brief summary of the program and its purpose should accompany all entries.

5 All entries will be treated in strict confidence.

6 Neither AMSOFT nor Amstrad can be held to be responsible for any loss or damage to any submission.

7 If you want the program returned, please supply a stamped self addressed envelope of adequate dimensions.

8 No correspondence can be entered in to concerning programs submitted for the cash prize competition.

9 The decision of the judges is final.

10 It is a condition of entry that all entrants have exclusive ownership of the copyright of the material submitted, and that the winner agrees to assign all copyright in the winning program to AMSTRAD. Where the entrant is more than one individual, then one person MUST be nominated and empowered to act on behalf of the entire group.
All entrants must undertake not to submit the same or a similar program to any other magazine, publisher or organisation until after the announcement of the winning entry.

11 AMSOFT may offer to publish programs either in CPC464 USER or as commercial software, in which case AMSOFT will agree the terms on an individual basis with the author(s) concerned. AMSOFT reserves the right to amend, alter or revise any program that it publishes.

12 No employees of AMSOFT or AMSTRAD, or their relatives may enter this competition.

See the entry form enclosed with this issue, and remember to fill it out with your name and address!

Page 21 ◊

Q. Anon.

Dear Sir,
I purchased a CPC464, but my Spectrum games won't load into it.

A.N.Other

A. Arrrrgh!

In common with most other proprietory computer systems, the CPC464 will only load software specifically designed for it.

Q. Overbaud?

Dear Sir,
I have just read the PCW review, and was very interested in the article about your CPC464 home computer. I have been waiting for about 9 months since I sold my last computer, for a computer like yours to be announced. It seems just the thing for my home user needs. But one detail I would like to know, what baud rate does the printer port run at ?
MM, Edgware

A. Yes, we have no bauds

The CPC464 printer port is a parallel port (centronics) where each character is sent individually at a rate governed by the printer. It does not therefore have a baud rate as such, as associated with serial interface systems.

Q. Various questions

Dear Sir,
One of the features that has most attracted me to the CPC464 is the excellence of the display - free from the jitters that plagues all modulator driven TVs. As a life-long migraine sufferer, these jitters must be avoided at all costs.

I also want to use the 80 column mode, which is not satisfactory with a low cost colour monitor, so in view of this I asked if I could buy a system with both monitors, and was told that the monitors are not presently supplied individually.

As a potential user and probable disk and printer customer as well, can you help me on this matter, please?

The length of cable between the monitor and keyboard is also rather too short for my purposes - is there any reason why additional leads should not be interposed?

The manual appears to be first class - but obviously not comprehensive. Are there any other publications available ?
JHW,Worcester

A. Various replies

1 The monitors are not presently supplied individually. The shortage of small screen TV tubes means that there are only enough to provide for the production of the computer units themselves. A number of other monitors are available on the market, although as you will have noticed, some cost as much a complete CPC464 system.

However, although the standard gold on blue display is not ideal for 80 column text, any combination of black on either Red, Green or Blue will be much crisper, since there is minimal mixing of adjacent phosphor dots on the screen display. Black text on a light blue or green background will be perfectly legible.

2 The voltage drop that occurs along an extension cable may affect the operation of the unit. It is not a task that should be undertaken by anyone other than a competent service engineer. However, you may use a separate power supply (such as that in the MP1), and then the video connector lead may be extended up to 10 feet or more without difficulty. The power output from the monitor should be left disconnected.

3 The advanced user publications are the concise BASIC specification (SOFT157) and the firmware specification (SOFT158). These are not user 'guides' as such, but are meticulously indexed and very accurate reference works which set out the rules for programmers. A series of more general publications are well under way from various authors, and we will obviously be reviewing them in this magazine. Some of the better ones may be offered to User club members at a special rate.

GENERAL

To conclude this marathon Q/A session for the CPC464 user, here's a list of the quickies:

Q. What language has the computer got?

A. A Microsoft-style BASIC with extensions for graphics, sound and interrupt handling.

Q. Which types of Joystick are compatible?

A. Any joystick with a standard 9-pin connector and DC switches should be compatible, although to operate the second joystick, the JY1 is required since this provides the socket for the second stick in its base.

■ CPC464 USER

AMSOFT'S Soft Centre....

A computer as comprehensive and attractive as the CPC464 will inevitably attract a wide range of software: educational, entertainment, utility, business, 'custom' applications of all types.

Amsoft has been established to encourage and harness this effort to provide software for CPC464 users.



Watch out for *AMSOFT* software centres at retailers -or buy from *AMSOFT* direct. The autumn catalogue is available now to help you choose, although you should also watch the press for details of the new software being launched each week.